



MANU0363-01 - Rev. A, September 28, 1998

Software Version 5.0.x

FORE Systems, Inc.

1000 FORE Drive Warrendale, PA 15086-7502 Phone: 724-742-4444

FAX: 724-742-7742

http://www.fore.com

#### Legal Notices

Copyright ° 1995-1998 FORE Systems, Inc. All rights reserved. FORE Systems, ForeRunner, ForeView, ForeThought, ForeRunnerLE, PowerHub and AVA are registered trademarks of FORE Systems, Inc. CellPath, EdgeRunner, Zero Hop Routing, MSC, TNX, Voice Plus, StreamRunner, Universal Port, ASN, CellStarter, Intelligent Infrastructure, I2, NetPro, FramePlus, ForeRunnerHE, ASX, Network Of Steel, and Networks Of Steel are trademarks of FORE Systems, Inc. All other brands or product names are trademarks of their respective holders.

**U.S. Government Restricted Rights.** If you are licensing the Software on behalf of the U.S. Government ("Government"), the following provisions apply to you. If the Software is supplied to the Department of Defense ("DoD"), it is classified as "Commercial Computer Software" under paragraph 252.227-7014 of the DoD Supplement to the Federal Acquisition Regulations ("DFARS") (or any successor regulations) and the Government is acquiring only the license rights granted herein (the license rights customarily provided to non-Government users). If the Software is supplied to any unit or agency of the Government other than DoD, it is classified as "Restricted Computer Software" and the Government's rights in the Software are defined in paragraph 52.227-19 of the Federal Acquisition Regulations ("FAR") (or any successor regulations) or, in the cases of NASA, in paragraph 18.52.227-86 of the NASA Supplement to the FAR (or any successor regulations).

Printed in the USA.

No part of this work covered by copyright may be reproduced in any form. Reproduction, adaptation, or translation without prior written permission is prohibited, except as allowed under the copyright laws.

This publication is provided by FORE Systems, Inc. "as-is" without warranty of any kind, either express or implied, including, but not limited to, the implied warranties or conditions of merchantability or fitness for a particular purpose. FORE Systems, Inc. shall not be liable for any errors or omissions which may occur in this publication, nor for incidental or consequential damages of any kind resulting from the furnishing, performance, or use of this publication.

Information published here is current or planned as of the date of publication of this document. Because we are improving and adding features to our products continuously, the information in this document is subject to change without notice.

RESTRICTED RIGHTS LEGEND. Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 (October 1988) and FAR 52.227-19 (June 1987).

## **List of Figures**

#### **List of Tables**

Pr	ef	а	C	е
----	----	---	---	---

		ariesort		
Typogra	aphical S	Styles	i	i
CHAP1		nation Indicators		٧
_			4 4	_
1.1		reView Interface		
1.2		ion with HP OpenView		
1.3		ery and Topology Mapping		
1.4		vitch Management via the Web Browser		
1.5	Access	to ForeView 4.3 Tools	1 - 7	7
	1.5.1	Upgrade Considerations	1 - 8	9
1.6	On-line	Help	1 - 9	9
СНАРТ	TER 2	Software Installation		
2.1	Unpack	ing Information	2 - 2	2
2.2	System	Requirements	2 - 3	3
	2.2.1	Hardware	2 - 3	3
		2.2.1.1 UNIX Requirements	2 - 3	3
		2.2.1.2 PC Requirements	2 - 3	3
	2.2.2	Software		
		2.2.2.1 UNIX Requirements		
		2.2.2.2 PC Requirements		
2.3	Obtainir	ng the Software Distribution		
	2.3.1	Obtaining the Software Distribution via TACtics Online		
	2.3.2	ForeView Licensing		
2.4	Before `	You Begin	2 - 7	7
	2.4.1	Using ForeView 4.3 Tools with ForeView 5.0		
	2.4.2	Removing ForeView 4.3 and Upgrading to ForeView 5.0		
	2.4.3	Retaining ForeView 4.3 Tools and Upgrading to ForeView 5.0		
	2.4.4	Removing Earlier ForeView Installations	2 - 9	2

2.5	Installa	tion from a TACtics Online File	10
	2.5.1	UNIX Platform	10
		2.5.1.1 Software Installation	10
	2.5.2	Windows NT Platform	
		2.5.2.1 Software Installation	11
2.6	Installa	tion from a CD-ROM	12
	2.6.1	UNIX Platform	12
	2.6.2	Windows NT	
		2.6.2.1 Import the Host Name File (If Necessary) 2 -	14
CHAP	TER 3	Getting Started	
3.1	Overvi	ew of <i>ForeView</i> Components	. 2
3.2	Runnir	g ForeView	. 3
	3.2.1	UNIX Commands for Daemons	. 4
	3.2.2	Using Windows NT Services	- 5
СНАР	TER 4	Network Discovery and Monitoring	
4.1	The Di	scovery NIAM (fvdisco)4 -	. 2
	4.1.1	How Fvdisco Discovers the Topology	- 2
		4.1.1.1 Discovery of LAN Access Devices	
		4.1.1.2 What Information fvdisco Discovers	- 3
		4.1.1.3 Seeding the Discovery	
		4.1.1.4 Fvseed Command-Line Arguments	
		4.1.1.5 Fvdisco Command-Line Arguments	. 5
4.2	The M	nitoring NIAM (fvmon)	- 6
	4.2.1	What Fvmon Does	. 6
	4.2.2	How Fvmon Works	
		4.2.2.1 How Fymon Determines the Status of Nodes 4	
		4.2.2.2 How Fymon Determines the Status of Links	
		4.2.2.3 How Fymon Adapts to Network Conditions	
		4.2.2.4 How Fymon Implements Tenacity	
		4.2.2.5 Status Records	
4.0		4.2.2.6 Fymon Command-Line Arguments	
4.3		preView Stores Database Information4 -	
	4.3.1	The Database Schema	
	4.3.2	Adding Non-FORE Devices to the Database	
		4.3.2.1 Fvaddlink Command-Line Arguments	
1 1	How 5	4.3.2.2 Running Fvaddlink from the OpenView Maps	
4.4		preView Integrates with HP OpenView	
	4.4.1	The ForeView Applications	
		4.4.1.1 The Fvtopomap Application	16

	4.4.1.2 The Fvovmon Application	
	4.4.1.3 The Fvtrapd Application	
	4.4.2 The Topology Mapping Process	
	4.4.2.1 Fvtopomap Arguments	
	4.4.3 The Database Synchronization Process	
	4.4.3.1 Fvovmon Command-Line Arguments	
	4.4.4 The Trap Handling Process	
	4.4.4.2 Supported Traps	
	4.4.4.3 Fytrapd Arguments	
	4.4.4.4 Trap Configuration in HP OpenView	
	4.4.4.4.1 Changing Trap Severity	
	4.4.4.4.2 Creating a Trap Category	
4.5	Changing Community Strings	4 - 27
4.6	Summary of ForeView Commands	4 - 28
CHAI	PTER 5 Network Mapping	
5.1	The ForeView ATM Maps	5 - 2
	5.1.1 Physical vs. Logical Topology	5 - 2
	5.1.2 Root Level Submap	
	5.1.3 ATM Networks Submap	5 - 4
	5.1.4 Switch Connections Submap	
	5.1.5 Fabrics Submap	
	5.1.6 Redundant NNI Links Submap	
5.2	Launching the ATM Switch Element Manager	
5.3	How Status Colors are Derived	5 - 10
5.4	Managing Symbols and Maps	5 - 11
	5.4.1 Creating Location Symbols	5 - 11
	5.4.2 How Location Symbols are Implemented	5 - 12
5.5	Retained Links When Pasting Edge Devices	5 - 13
5.6	Note About the Unmanage Objects Command	5 - 14
APPE	ENDIX A Tips and Troubleshooting	
A.1	Checking ForeView Processes	A - 1
	A.1.1 Checking Fvtopomap	A - 2
	A.1.1.1 Checking Fvtopomap on UNIX	
	A.1.1.2 Checking Fvtopomap on Windows NT	
	A.1.2 Checking Error Logs	
	A.1.3 Examining the OVW Database	
	A.1.4 Deleting <i>ForeView</i> Map Objects	
A.2	Use Netscape to View On-line Help	A - 6

A.3	Mainta	aining the <i>ForeView</i> Database	7
	A.3.1	Removing Old Status Entries from the Database	7
A.4	Modify	ring Application Launch Parameters A	8
	A.4.1	The Windows NT Launcher A	
		A.4.1.1 Example	
	A.4.2	The UNIX Boot Script	- 10
APPE	NDIX B	The ForeView Database Schema	
B.1	Schem	na Overview	- 2
B.2	Topolo	gy Data Tables	- 5
	B.2.1	The fv node 1 Table	
	B.2.2	The fv container 1 Table	
	B.2.3	The fv_address_1 Table	- 9
	B.2.4	The fv_interface_1 Table	- 10
	B.2.5	The fv_endpoint_1 Table	- 11
	B.2.6	The fv_link_1 Table	
	B.2.7	The fv_sig_entity_1 Table	
	B.2.8	The fv_sig_vpi_1 Table	
	B.2.9	Supporting Tables	
		B.2.9.1 The fv_container_t_1 Table	
		B.2.9.2 The fv_address_t_1 Table	
		B.2.9.3 The fv_address_src_1 Table	
		B.2.9.4 The fv_interface_t_1 Table	
		B.2.9.5 The fv_link_t_1 Table	
		B.2.9.6 The fv_sig_proto_t_1 Table	
		B.2.9.7 The fv_sysoid_1 Table	
D 0	N 4 :4 -	·	
B.3		oring Data Tables	
	B.3.1	The fv_node_s_1 Table	
	B.3.2	The fv_hwc_link_s_1 Table	
	B.3.3	The fv_uni_link_s_1 Table	
	B.3.4	The fv_spans_link_s_1 Table	
	B.3.5	Supporting Tables	
		B.3.5.1 The fv_mon_interval Table	
		B.3.5.2 The fv_mon_priority_1 Table	
D 4		_ <i>,</i>	
B.4		ng Data Tables	
	B.4.1	The fv_mapping_1 Table	
	B.4.2	Supporting Tables	
		B.4.2.1 The fv_table_t_1 Table	
B.5	SNMP	Configuration Data Table	- 25

	B.5.1	The fv_snmp_conf_1 Table
B.6	Queryin	g the DatabaseB - 26
	B.6.1	Query for Discovered Nodes
	B.6.2	Query for the Number of Monitored NodesB - 26
	B.6.3	Query for the Number of Links
	B.6.4	Query for the Number of NNI/UNI/HWC Links
	B.6.5	Query to List the Links by Port and IP Address
	B.6.6	Query for the Status of the Links
	B.6.7	Table DumpsB - 30
APPE		Licensing Agreements
C.1	Java(tm	) Runtime Environment
C.2	Apache	Group
C.3	JGL - T	ne Generic Collection Library for Java(tm)
C.4	PERL S	cript
C.5	ODBC I	Oriver Manager License
Acror	nyms	

Glossary

Index

# **List of Figures**

CHAPTER 1	Introduction
Figure 1.1	A ForeView Submap1 - 2
Figure 1.2	FORE's Web-embedded Element Manager
Figure 1.3	Element Manager's ATM Switch Front Panel
Figure 1.4	On-line Help Example1 - 9
CHAPTER 2	Software Installation
Figure 2.1	FORE's TACtics WWW Page2 - 5
CHAPTER 3	Getting Started
Figure 3.1	ForeView and OpenView Integrated Processes
Figure 3.2	Windows NT Services Dialog
CHAPTER 4	Network Discovery and Monitoring
Figure 4.1	The ForeView NIAMs Interface with the Network
Figure 4.2	The ForeView Database Stores Node, Link, and Status Information4 - 11
Figure 4.3	The Fvaddlink Dialog
Figure 4.4	Fvtopomap Creates the OpenView ATM Map4 - 18
Figure 4.5	Fvovmon Synchronizes ovwdb with fvdb 4 - 20
Figure 4.6	Fvtrapd Handles ATM Switch Traps
CHAPTER 5	Network Mapping
Figure 5.1	Level 1 Submap - Root Level
Figure 5.2	Level 2 Submap - ATM Networks
Figure 5.3	Level 3 Submap - Switch Connections
Figure 5.4	Switch Fabrics Submap5 - 6
Figure 5.5	Element Manager Login Screen
Figure 5.6	Element Manager's ATM Switch Front Panel
Figure 5.7	ForeView: Manage Objects Command5 - 14
APPENDIX A	Tips and Troubleshooting
Figure A.1	Windows NT Task Manager
APPENDIX B	The ForeView Database Schema
Figure B.1	Example of the Database Relationship SchemaB - 4

### List of Figures

APPENDIX C	Licensing Agreements
Figure B.4	Mapping Relations in the ForeView Database
Figure B.3	Status Monitoring Relations in the ForeView Database B - 17
Figure B.2	Topology Relations in the ForeView Database

## **List of Tables**

CHAPTER 1	Introduction	
CHAPTER 2	Software Installation	
CHAPTER 3 Table 3.1 Table 3.2	Getting Started Start /Stop/Status Commands for UNIX	
CHAPTER 4 Table 4.1 Table 4.2	Network Discovery and Monitoring Link Status Summary	
CHAPTER 5	Network Mapping	
APPENDIX A	Tips and Troubleshooting	
APPENDIX B	The ForeView Database Schema	
Table B.1	Summary of ForeView Data Tables	B - 1
Table B.2	fv_node_1	B-8
Table 5.1	fv_container_1	B-9
Table B.3	fv_address_1	B-9
Table B.4	fv_interface_1	B - 10
Table B.5	fv_endpoint_1	B - 1′
Table B.6	fv_link_1	B - 1′
Table B.7	fv_sig_entity_1	
Table B.8	fv_sig_vpi_1	B - 12
Table B.9	fv_container_t_1	B - 13
Table B.10	fv_address_t_1	B - 13
Table B.11	fv_address_src_1	B - 14
Table B.12	fv_interface_t_1	B - 14
Table B.13	fv_link_t_1	B - 14
Table B.14	fv_sig_proto_t_1	B - 15
Table B.15	fv_sysoid_1	B - 15
Table B.16	MIB Variables for Unique ID	B - 16
Table B.17	fv_node_s_1	B - 18

#### List of Tables

Table B.18	fv_hwc_link_s_1	B - '	19
Table B.19	fv_uni_link_s_1	B - 2	20
Table B.20	fv_spans_link_s_1	B - 2	21
Table B.21	fv_mon_interval	B - 2	21
Table B.22	fv_mon_priority_1	B - 2	22
Table B.23	fv_entity_t_1	B - 2	22
Table B.24	fv_mapping_1	B - 2	23
Table B.25	fv_table_t_1	B - 2	24
Table B.26	fv_snmp_conf_1	B - 2	25

## **APPENDIX C** Licensing Agreements

## **Preface**

The intent of this manual is to supply users of the *ForeView*® Network Management software with all the necessary information to successfully install and operate the *ForeView* software package. This document was created for users with various levels of experience. If you have any questions or problems with the installation, please contact FORE Systems' Technical Support.

## **Chapter Summaries**

**Chapter 1 - Introduction -** Provides an overview of the *ForeView* Network Management package, a description of hardware and software requirements, and general information pertaining to the capabilities and features of the *ForeView* Network Management software package.

**Chapter 2 - Software Installation -** Guides you through the installation of your *ForeView* Network Management software.

**Chapter 3 - Getting Started -** Explains *ForeView's* new architecture and defines the processes and applications that enable *ForeView* to interface with the ATM network. Also explains how to start *ForeView's* discovery and monitoring mechanisms.

**Chapter 4 - Network Discovery and Monitoring -** Explains how *ForeView* utilizes a discovery and status processes to discover FORE Systems switches, NNI and UNI links, and the endpoints of NNI and UNI links in an ATM network.

**Chapter 5 - Network Mapping -** Explains how *ForeView* utilizes a mapping utility to create an OpenView map that represents the near-physical topology of your ATM network.

**Appendix A - Tips and Troubleshooting** - Provides usage tips and troubleshooting information for *ForeView*.

**Appendix B - The ForeView Database Schema** - Provides a description of the database schema, tables, and query examples.

**Appendix C - Licensing Agreements** - Provides licensing statements for some third-party software used in *ForeView* 5.0.

## **Technical Support**

In the U.S.A., you can contact FORE Systems' Technical Assistance Center (TAC) using any one of the following methods:

1. You can receive on-line support via TACtics Online at:

#### http://www.fore.com/tac

2. You can contact Technical Support via e-mail at:

#### support@fore.com

3. You can telephone your questions to Technical Support at:

4. You can FAX your questions to Technical Support at:

#### +1 724-742-7900

Technical support for non-U.S.A. customers should be handled through your local distributor.

No matter which method is used for support, please be prepared to provide your support contract ID number, the serial number(s) of the product(s), and as much information as possible describing your problem/question.

## **Typographical Styles**

Throughout this manual, specific commands to be entered by the user appear on a separate line in bold typeface. In addition, use of the Enter or Return key is represented as <ENTER>. The following example demonstrates this convention:

cd /usr <ENTER>

Commands or file names that appear within the text of this manual are represented in the following style: "...the fore\_install program will install this distribution"

As in the following example, any messages appearing on your screen during software installation and network interface administration will appear in Courier font to distinguish them from the rest of the text.

.... Are all four conditions true?

## **Important Information Indicators**

To call your attention to important information that must be reviewed to ensure correct and complete installation, as well as to avoid problems with your software, FORE Systems utilizes the following *CAUTION/NOTE* indicators.

Information contained in **CAUTION** statements is important for proper installation/operation. **CAUTION** statements can prevent possible equipment damage and/or loss of data and will be indicated as:

#### **CAUTION**



You risk damaging your equipment and/or software if you do not follow these instructions.

Information contained in **NOTE** statements has been found important enough to be called to the special attention of the operator and will be set off from the text as follows:



FORE Systems strongly recommends that you disconnect the serial cable once you have configured the *ForeRunner* switch and then access the *ForeRunner* switch over the ATM network.

# CHAPTER 1

## Introduction

ForeView Network Management is a sophisticated management application that integrates with HP OpenView on the UNIX (Solaris) and Windows NT platforms to provide management of FORE ATM networks. ForeView is a graphical, end-to-end network management application for configuring, monitoring, and controlling a FORE Systems' ATM-switched network infrastructure.

The key features of *ForeView* include:

- An intuitive Graphical User Interface (GUI) that allows network administrators to monitor, configure, analyze, troubleshoot, and proactively manage the ATM network.
- Integration with HP OpenView, a recognized leader in network management software.
- New architecture and an underlying database for discovery and status monitoring.
- The ability to launch FORE's new Web-embedded ATM switch managers.
- Access to management tools provided by the ForeView 4.3 release.
- On-line help in HTML format.

### 1.1 The ForeView Interface

*ForeView*'s intuitive GUI provides network administrators with a quick visual assessment of the general state of FORE devices, as well as detailed examination of connection status, and diagnosis in the event of errors. Figure 1.1 illustrates how a particular ATM network might be mapped. See Chapter 5 for more information about *ForeView*'s network mapping capabilities.

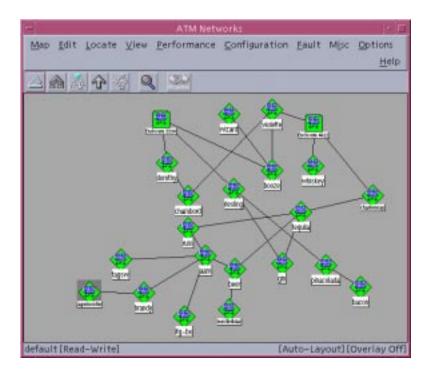


Figure 1.1 - A ForeView Submap

## 1.2 Integration with HP OpenView

ForeView Network Management software manages FORE ATM networks in a variety of network environments. Although ForeView manages the ATM network using the IP protocol, it does not manage the IP network itself. Those tasks are left to HP OpenView. ForeView is designed to work integrally with HP OpenView, under native operating systems Solaris and Windows NT, to manage FORE Systems' devices.

Because *ForeView* is integrated with HP OpenView, the network manager can take advantage of some of the features of the NMS, such as topology management, event logging, error trapping, graphing modules, and threshold programming.

HP OpenView provides a management application called Network Node Manager (NNM) for use in managing TCP/IP networks and network devices that support SNMP. This application runs under the HP OpenView Windows (OVW) graphical user interface. HP OpenView provides configuration, performance, and fault management support for multi-vendor networks.

HP OpenView provides topology maps (see Figure 1.1) based on the ATM network information discovered by *ForeView*. The maps provide a graphical and hierarchical representation of the ATM network.

## 1.3 Discovery and Topology Mapping

*ForeView* discovers and maps ATM networks to create a campus view of interconnected FORE switches and devices. Because ATM is connection-oriented, *ForeView* displays topology information representative of the physical network. *ForeView* uses ATM signalling (UNI 3.x/4.x and/or SPANS) to discover and map any UNI 3.x/4.x or SPANS compliant device and related connections. The data is then displayed in specific map views.

*ForeView* uses a persistent database architecture for discovery and status monitoring, providing for scalability and true distributed network management. A network administrator can customize the discovery and status polling capabilities of *ForeView* to effectively manage networks with hundreds of switches.



There are inherent dependencies on the version of the *ForeThought* software running on the ATM switch. This release of *ForeView* will support *ForeThought* versions from 5.0 to 5.3.

## 1.4 ATM Switch Management via the Web Browser

This release of *ForeView* provides access to a new method of ATM switch configuration and management through an easy-to-use World Wide Web (WWW) interface using a Web browser such as Netscape or Microsoft's Internet Explorer. The WWW provides a graphical interface with a simple structure that makes use of internet presentation attributes common in the HTML (Hypertext Markup Language) environment such as pages, frames, and tables. The WWW interface, called an Element Manager, mirrors FORE's command line user interface for ATM switch configuration and management.

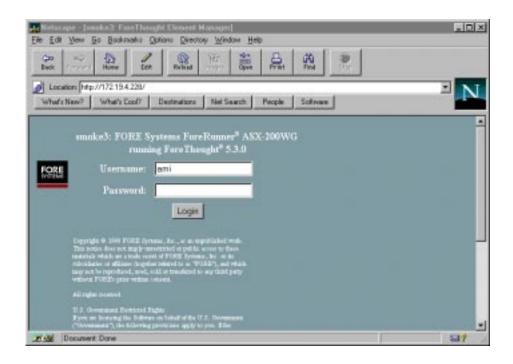
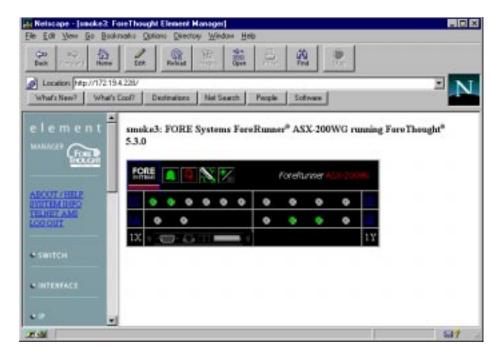


Figure 1.2 - FORE's Web-embedded Element Manager

The Element Manager provides a graphical representation of an actual FORE Systems' ATM switch, including the number and type of network modules installed in the device and the status of the ports on each of these modules.



**Figure 1.3 -** Element Manager's ATM Switch Front Panel

### 1.5 Access to ForeView 4.3 Tools

*ForeView* 5.0 (this release) is built using a different software architecture than earlier versions. Built on an underlying integrated database, *ForeView* 5.0 migrates tools that were provided previously by *ForeView* 4.3 and makes them available through the HP OpenView menu system.

The following ForeView 4.3 tools can co-exist with ForeView 5.0:

- Configure Path and Channel (fvchan)
- Trace Path and Channel (fvtracer)
- Log (fvlog)
- Graph (fvgraph)
- VLAN Manager (fvlan)
- AMI Interface with Switch
- ForeView RMON ST
- OAM Cell Monitor
- Call and Performance Records
- Front panels for ES-3810s, *PowerHubs*, and FORE ATM switches running software other than *ForeThought* 5.2.

## 1.5.1 Upgrade Considerations

For customers currently running *ForeView* 4.3, there are two upgrade options available:

- Removing *ForeView 4.3* and installing *ForeView 5.0*.
- Retaining some of the *ForeView* 4.3 tools and upgrading to *ForeView* 5.0.

Customers running *ForeView* 4.3 software who are not interested in maintaining the use of *ForeView* 4.3 tools should refer to "Removing ForeView 4.3 and Upgrading to ForeView 5.0" on page 2 - 8.

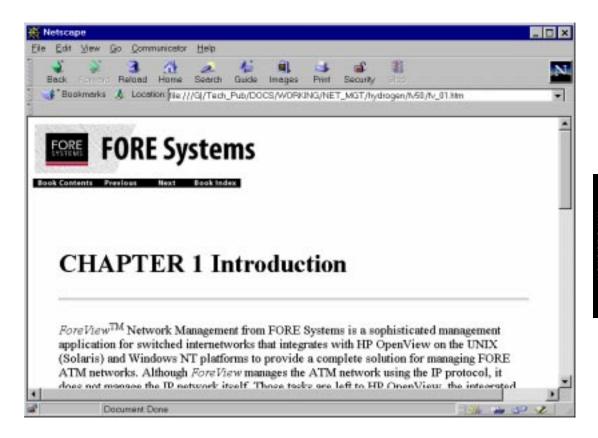
Customers running *ForeView* 4.3 software who want to integrate the *ForeView* 4.3 tools so they can be launched from *ForeView* 5.0 should refer to "Retaining ForeView 4.3 Tools and Upgrading to ForeView 5.0" on page 2 - 8.



No previous versions of *ForeView*, other than *ForeView 4.3*, interoperate with *ForeView 5.0*.

## 1.6 On-line Help

ForeView provides an on-line Help manual in HTML format to assist you during network management tasks. You can get on-line help by selecting the ForeView option under the OpenView Help menu. A display similar to Figure 1.4 provides help related to the task you are performing.



**Figure 1.4 -** On-line Help Example

#### Introduction

# CHAPTER 2

## **Software Installation**

This chapter is designed to guide you through the installation of the *ForeView* network management software. The procedures contain step-by-step instructions for the successful installation of the software. This chapter includes:

- Information on system requirements (both hardware and software), found on page 2-3.
- How to obtain *ForeView* software via FORE's TACtics Online web site, including license key information, found on page 2-5.
- Information on how *ForeView* 5.0 interoperates with earlier versions, including how to upgrade to *ForeView* 5.0 and remove *ForeView* 4.3, on page 2-8; and how to upgrade to *ForeView* 5.0 and integrate *ForeView* 4.3 tools, on page 2-8.
- Installation instructions (for UNIX and Windows NT) for downloaded software, on page 2-10; and from a CD-ROM distribution, on page 2-12.

It is strongly suggested that you read all of this information carefully before attempting installation. In addition, check the Release Notes that come with your software for the most current information and pre-requisites.



ForeView must be installed after HP OpenView is installed.

## 2.1 Unpacking Information

Upon receipt of, and before opening, your *ForeView* Network Management software, inspect the package for any damage that may have occurred during shipping. If the package shows any signs of external damage or rough handling, notify your carrier's representative.

When unpacking the *ForeView* Network Management software be sure to keep all original packing materials. They may be needed for return of the product.

**CAUTION** 



All products returned to FORE Systems, under warranty, must be packed in their original packing materials.

## 2.2 System Requirements

Before installing *ForeView* network management software, be sure you have a platform which meets the following requirements. You will need to have the root password for the machine on which you wish to install *ForeView*.

#### 2.2.1 Hardware

### 2.2.1.1 UNIX Requirements

- SPARCstation (UltraSparc 1 or SPARCstation 10 Recommended)
- Ethernet interface or ATM 200E or ForeRunnerHE network interface
- Color monitor
- Free Disk Space: 150MB (minimum)
- Swap Space (w/OV): 200 MB (recommended)
- Memory: 64MB (minimum) (128 MB recommended)

#### 2.2.1.2 PC Requirements

- IBM-compatible PC with a Pentium II CPU
- Ethernet interface or ATM 200E, ForeRunnerLE, or ForeRunnerHE network interface
- Color monitor (1024 x 768, 256 colors)
- Free Disk Space: 120 MB (minimum)
- Memory: 128 MB (minimum)

#### 2.2.2 Software

#### 2.2.2.1 UNIX Requirements

• Solaris 2.5.1 or 2.6 (version 2.6 recommended).

If running Solaris 2.5.1, you must have installed the recommended Solaris 2.5.1 patch cluster. The patch cluster is available at:

#### http://sunsolve.sun.com

- HP OpenView SNMP Platform or NNM Version 5.0.1
- HP OpenView 5.01 consolidated patch PSOV\_01688 or later (for your convenience, this patch is included on the ForeView CD)
- Netscape Navigator 3.0 or later
- Prior to installing ForeView, the Informix software package must not be installed on the workstation.

#### 2.2.2.2 PC Requirements

Microsoft Windows NT 4.0 and Service Pack 3



The file system in which *ForeView* will be installed must be an NTFS partition to support long file names used by *ForeView*.

- Virtual Memory (Paging File Size): 200 MB minimum
- HP OpenView 5.0.2
- Netscape Navigator 3.0 (or later) or Microsoft Internet Explorer (MSIE) 4.0
- Prior to installing *ForeView*, the following software packages must *not* be installed on the workstation:
  - Informix
  - Redpoint SnmpQL
  - Trifox Vortex

## 2.3 Obtaining the Software Distribution

Before beginning the installation process, you will need the software distribution from FORE Systems. The software can be obtained via the WWW at TACtics Online or CD-ROM.

- To obtain the file via TACtics Online, you must have an account.
- To obtain the file from CD-ROM, you must have the distribution CD-ROM from FORE Systems.

## 2.3.1 Obtaining the Software Distribution via TACtics Online

You may obtain the latest software via the TACtics Online web link. In the U.S.A., customers can reach FORE Systems' TACtics Online using the following URL to connect to FORE's World Wide Web site:

http://www.fore.com/tac/



Figure 2.1 - FORE's TACtics WWW Page

The latest technical documentation for this software (and other FORE products) also is available from TACtics Online, as well as the FORE Technical Manuals web link:

http://www.fore.com/products/manuals.htm

## 2.3.2 ForeView Licensing

*ForeView* 5.0 requires the user to enter a permanent license key. This key should be received with the software distribution CD-ROM from FORE Systems. If your software was downloaded from TACtics Online, a valid permanent license key will be e-mailed to you.



The *ForeView* 4.3 license CANNOT be used to install *ForeView* 5.0.

## 2.4 Before You Begin

Before you begin the *ForeView* installation, make sure you have the following:

- The ForeView distribution, either via TACtics Online download or CD-ROM with installation instructions.
- A valid *ForeView* license key number. If you need a license key, contact the FORE Systems TAC, as described in "Technical Support" on page ii of the Preface.
- The IP address or hostname of at least one ATM switch in your network to use as a "seed switch" to enable network discovery. The *ForeView* workstation must be able to reach (ping) this switch via IP and have SNMP read access.

## 2.4.1 Using ForeView 4.3 Tools with ForeView 5.0

*ForeView* 5.0 has been re-designed using a completely new software architecture. Depending on the installation option chosen, some management tools that were available in *ForeView* 4.3 can co-exist with *ForeView* 5.0. If you are currently running *ForeView* 4.3, there are two upgrade options available:

- Removing *ForeView* 4.3 and installing *ForeView* 5.0.
- Retaining some of the ForeView 4.3 tools and upgrading to ForeView 5.0.

The following ForeView 4.3 tools can co-exist with ForeView 5.0:

- Configure Path and Channel (fvchan)
- Trace Path and Channel (fvtracer)
- Log (fvlog)
- Graph (fvgraph)
- VLAN Manager (fvlan)
- AMI Interface with Switch
- ForeView RMON ST
- OAM Cell Monitor
- Call and Performance Records
- Front panels for ES-3810s, *PowerHubs*, and FORE ATM switches running software other than *ForeThought* 5.2.

These entries appear under the ForeView 4.3 menu on the OpenView menu bar.

## 2.4.2 Removing ForeView 4.3 and Upgrading to ForeView 5.0

Customers running *ForeView 4.3* software who are not interested in maintaining the use of *ForeView 4.3* tools need to do the following:

- 1. Remove *ForeView* 4.3 (described in "Removing Earlier ForeView Installations" on page 2-9).
- 2. Remove HP OpenView 4.x, if it is installed.
- 3. Install HP OpenView 5.x.
- 4. Install ForeView 5.0.

# 2.4.3 Retaining *ForeView* 4.3 Tools and Upgrading to *ForeView* 5.0

Customers running *ForeView* 4.3 software who want to retain the *ForeView* 4.3 tools so they can be launched from *ForeView* 5.0 need to do the following:

- 1. In UNIX, remove HP OpenView 4.x, if it is installed.
- 2. Install HP OpenView 5.x.
- 3. Install ForeView 5.0.
- 4. In UNIX, ensure that \$FOREVIEW\_HOME and \$FOREVIEW\_TOP are correctly set. (In Windows NT, FOREVIEW HOME is set in the registry.)
- 5. Run the fv43ovlink utility:
  - For UNIX, FOREVIEW\_TOP>/bin/config/fv43ovlink.
  - For Windows NT, \<FOREVIEW\_TOP>\bin\config\fv43ovlink.exe.

Note the use of FOREVIEW\_TOP to define the root of the *ForeView* 5.0 installation. Because FOREVIEW\_HOME was used for the previous *ForeView* releases, the use of FOREVIEW\_TOP allows for the co-existence of versions 5.0 and 4.3 on the same system and allows the user to launch a number of the *ForeView* 4.3 tools from the OpenView ATM map.



Remember that no previous versions of *ForeView*, other than *ForeView 4.3*, interoperate with *ForeView 5.0*.

## 2.4.4 Removing Earlier ForeView Installations

If you choose to remove earlier versions of *ForeView* before upgrading to *ForeView* 5.0, use the following procedures.

#### For UNIX

- 1. Exit all *ForeView* applications.
- 2. Exit HP OpenView.
- 3. If *ForeView* is installed in an alternate directory, make sure that the environment variable FOREVIEW\_HOME is set.



If *ForeView* is installed in an alternate directory it is imperative that you set the FOREVIEW\_HOME environment variable *before* you begin the removal to avoid problems during the procedure.

- 4. Log in as root.
- 5. Issue the following commands:

/bin/sh /usr/fore/foreview/install/remove.sh

rm -rf /usr/fore/foreview

#### For Windows NT

- 1. Exit all *ForeView* applications.
- 2. Exit HP OpenView.
- 3. In the Registry, delete the value FOREVIEW\_HOME under:

HKEY\_LOCAL\_MACHINE\Software\Fore Systems, Inc.

- 4. Delete the entire subdirectory associated with the FOREVIEW\_HOME Registry (for example, C:\ForeView).
- 5. Delete the ForeView directory under:

<System Drive>\<System Root>\Profiles\All Users\Start Menu\Programs

### 2.5 Installation from a TACtics Online File

#### 2.5.1 UNIX Platform

If you have retrieved a software file with a .Z extension, then you need to uncompress the file using the following command:

uncompress <filename>

where *<filename>* represents the full name of the software file you have retrieved. For example:

uncompress fv50\_sol.tar.Z



After the software distribution is uncompressed, do NOT untar the file. The install procedure will expect the software distribution to be in tarfile format.

#### 2.5.1.1 Software Installation

- 1. To install *ForeView* software after uncompressing the tar file, change to the /tmp directory by entering cd /tmp.
- 2. Extract the installation script by entering tar -xvf fv50\_sol.tar.
- Enter cd /solaris.
- 4. Run the following command to begin installation: ./install\_foreview.
- 5. Answer the questions that follow to complete the installation.

#### 2.5.2 Windows NT Platform

If you have retrieved a software file with a .zip extension, then you need to uncompress the files into a temporary directory using the following command:

where <filename> represents the full name of the software file you have retrieved. For example:

#### 2.5.2.1 Software Installation

- 1. To install *ForeView* software after uncompressing the zip file, change to the temporary directory containing the extracted files.
- 2. Run the following command to begin installation: setup.
- 3. Answer the questions that follow to complete the installation.

## 2.6 Installation from a CD-ROM

#### 2.6.1 UNIX Platform

Use the following procedure to install *ForeView* on a UNIX platform:

- Log in as root.
- 2. Insert the *ForeView* Network Management 5.0 CD-ROM into the CD-ROM drive and mount the file system as /cdrom. See your system's user guide for instructions on mounting CD-ROMs.
- 3. Change directory to /cdrom/solaris.
- 4. Run the following command to begin the installation:
  - ./install foreview
- 5. Answer the questions that follow to complete the installation.

#### 2.6.2 Windows NT

Use the following procedure to install *ForeView* on the Windows NT platform:

1. Insert the *ForeView* Network Management 5.0 CD-ROM into the CD-ROM drive.



If autorun is enabled, the installation will begin automatically. Proceed to step 5.

- 2. From the Windows NT Explorer, select the CD-ROM drive.
- 3. Select the Winnt directory on the CD-ROM.
- 4. Double-click Setup.exe from the list of files in the Winnt directory.
- 5. Continue with the installation by following the screen prompts.

#### 2.6.2.1 Import the Host Name File (If Necessary)

If the Windows NT workstation is not participating in a network-wide DNS system, the hosts file must contain the IP address to name mapping in order for *ForeView* to display switches by name rather than IP address. This file can be imported from a UNIX workstation:

1. Enter the following command:

ypcat hosts > hosts

2. Copy this file, via ftp, to the drivers directory on your local machine, such as:

\winnt\system32\drivers\etc

Otherwise, edit the file manually. Refer to Windows NT's on-line help for a description of the hosts file format.

## CHAPTER 3

## **Getting Started**

This chapter provides a brief overview of the *ForeView* architecture, explaining how *ForeView* interfaces with the ATM network, stores information in a database, and integrates with Open-View. *ForeView* uses Network Information Access Modules (NIAMs) and other applications to discover and monitor the ATM network. These processes pass information to the *ForeView* database and OpenView database and map as required. An overview of these integrated processes is shown in Figure 3.1.

Also in this chapter, the section "Running *ForeView*" page 3-3 includes information on how to run *ForeView*, including how to start, stop, and check the status of *ForeView* discovery and monitoring.

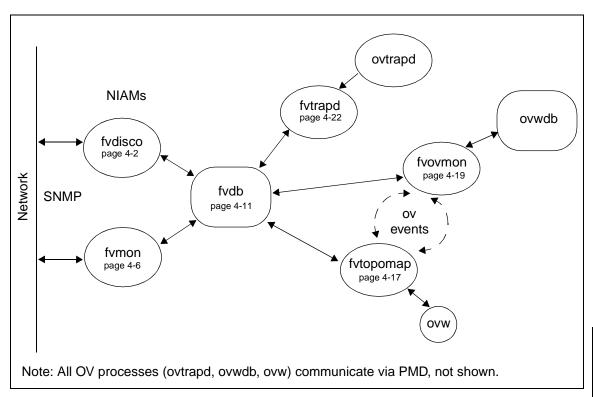


Figure 3.1 - ForeView and OpenView Integrated Processes

## 3.1 Overview of *ForeView* Components

The discovery and monitoring activities use the following *ForeView* components:

NIAMs (fvdisco and fvmon)

The NIAMs perform SNMP queries to the network to retrieve information from the actual ATM devices in the network. They act as the *ForeView* interface to the network, inserting information about the network into the *ForeView* database.

The discovery NIAM *fvdisco* discovers the topology of FORE ATM networks, including ATM switches, LAN Access devices, ATM-connected hosts, NNI and UNI links in an ATM network.

The status monitoring NIAM *fvmon* is responsible for monitoring and maintaining the status of discovered nodes and links.

ForeView Database (fvdb)

The NIAMs deposit information in the *ForeView* database. This is a relational database with tables containing information about all discovered nodes and links in the network.

ForeView Applications (fvovmon fvtopomap fvtrapd) The applications provide *ForeView*'s integration with OpenView. The applications access information in the *ForeView* database and pass it to OpenView applications to ensure that the OpenView ATM map is kept current and accurate.

- *fvovmon* synchronizes the OpenView database (*ovwdb*) with the *ForeView* database.
- fvtopomap creates an OpenView ATM submap showing the topology of the discovered ATM network.
- fvtrapd receives and handles FORE ATM switch traps forwarded by OpenView's ovtrapd.

## 3.2 Running ForeView

To run *ForeView* on either a Sun or Windows NT system, make sure the HP OpenView directory is the search path:

- /opt/OV/bin on UNIX
- <directory>\openview\bin on Windows NT

If this has not been done, modify the appropriate system configuration files.

Next, run OpenView:

- On UNIX, type the command ovw at the system prompt.
- On Windows NT, select the Start | Programs | HP OpenView | Network Node Manager menu command.

During the initial installation, the user is prompted to start both *ForeView* NIAMs, discovery NIAM *fvdisco* and the monitoring NIAM *fvmon*. Additionally, *fvdisco* and *fvmon* are configured to start automatically at system boot. Both are implemented in UNIX as daemons, and on Windows NT as Windows NT services.



To verify that *fvdisco* and *fvmon* are running, use **fvstatus** (UNIX) or check the Windows NT Services control panel.

#### 3.2.1 UNIX Commands for Daemons

Table 3.1 lists the UNIX commands to start and stop the NIAMs, as well as to check their status.

**Table 3.1 - Start / Stop / Status Commands for UNIX** 

UNIX Command	Description
fvstart [fvdisco fvmon]	Starts the daemons.  You can specify the discovery or monitoring NIAM. If you specify neither, both are started. On UNIX you also can include the <i>fvmon</i> command-line arguments, described in "Fvmon Command-Line Arguments" on page 4-10.
fvstop [fvdisco fvmon]	Stops the daemons.  You can specify the discovery or monitoring NIAM. If you specify neither, both are stopped. On UNIX you also can include the <i>fvmon</i> command-line arguments, described in "Fvmon Command-Line Arguments" on page 4-10.
fvstatus [fvdisco fvmon]	Displays the status of the daemons.



More information on the discovery and monitoring processes can be found in Chapter 4 of this manual, including specific flags related to discovery and monitoring.

## 3.2.2 Using Windows NT Services

Because both *ForeView* NIAMs are registered as services on Windows NT, you can use the Windows NT Services control panel to start, stop, pause, or continue the services. View the control panel from Start\Settings\Control Panel\Services to see the current status and startup mode of the services.

The Service list in Figure 3.2 shows ForeView Discovery (*fvdisco*) and ForeView Status Monitor (*fvmon*). The startup mode (Automatic, Manual, or Disabled) can be selected via the Startup... button.

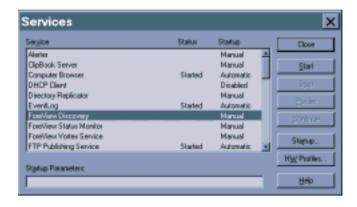


Figure 3.2 - Windows NT Services Dialog

Table 3.2 lists the commands to start and stop the NIAMs, as well as to check their status for Windows NT, although it is more typical to use the Windows NT services control panel.

**Table 3.2 - Start /Stop/Status Commands For Windows NT** 

Windows NT Command	Description
fvsrvc -start	Starts the services.
[fvdisco, fvmon]	You can specify the discovery or monitoring NIAM. If you specify neither, both are started.
fvsrvc -stop	Stops the services.
[fvdisco, fvmon]	You can specify the discovery or monitoring NIAM. If you specify neither, both are stopped.
fvsrvc -status	Displays the status of the services.

## Getting Started



# **Network Discovery and Monitoring**

*ForeView* uses two Network Information Access Modules (NIAMs) to interface with the ATM network. The NIAMs, *fvdisco* and *fvmon*, are background processes that issue SNMP queries to discover and then monitor the nodes and links in the network. The NIAMs insert information about the nodes and links into the tables of the *ForeView* database (*fvdb*).

The distributed, modular architecture of the NIAMs allow feature independence and scalability for this and future releases of *ForeView*.

Other *ForeView* applications retrieve information from *fvdb* and pass that information to Open-View, so that an accurate and current representation of the ATM network appears in the Open-View maps.

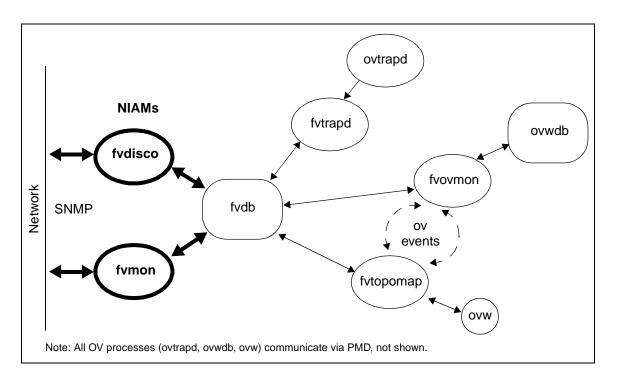


Figure 4.1 - The Fore View NIAMs Interface with the Network

## 4.1 The Discovery NIAM (fvdisco)

The Discovery NIAM (*fvdisco*) discovers the topology of a *ForeRunner* ATM network and populates the *ForeView* database with that information.

## 4.1.1 How *Fvdisco* Discovers the Topology

The *fvdisco* NIAM discovers the topology of the network by walking the MIBs on *ForeRunner* ATM switches. The MIBs contain all information that *ForeView* requires.

The *fvdisco* NIAM first discovers the seed switches, specified when *ForeView* was installed. It then walks the signalling MIBs on these switches. From these tables it gathers the IP addresses of neighboring devices. It then proceeds to walk the MIBs on the seed switches' neighbors, eventually discovering all contiguous ATM switches in the cloud.

Only switches that can be reached via SNMP are discovered and added to the *ForeView* database. If *ForeView* cannot read a switch SNMP MIB, the switch is not added. Discovered switches that are added to the database become the seed switches for the next discovery cycle.



There are inherent dependencies on the version of the *ForeThought* software running on the ATM switch. This release of *ForeView* will support *ForeThought* versions from 5.0 to 5.3.

#### 4.1.1.1 Discovery of LAN Access Devices

FORE Systems access devices (ES-3810s, etc.) are discovered by proxy via the uplink to an ATM switch. Only access devices that are ATM connected are discovered. The discovery NIAM uses the ILMI SNMP Proxy (ISP) MIB on the ATM switch to query the access device. It does not directly query the SNMP agent on the LAN Access devices.

ForeView must have the write community string for the switch in order to discover LAN access devices connected to the switch.



The discovery of LAN access devices is dependent on the device's capability to honor ILMI SNMP get requests.

#### 4.1.1.2 What Information fydisco Discovers

It discovers the following components in an ATM network:

- ATM switches
- PowerHub 7000s, 8000s, ASN-9000s and ES-3810s
- NNI links
- UNI links.

Device information includes the following data:

- Device name
- Port information
- IP address
- Device type and subtype (based on the system object identifier)
- Enclosure identifier
- Board number (for switch fabrics)
- Connectivity information, including all link statuses
- Signalling path information (both SPANS and UNI 3.x/4.x).

This information is made available to its clients (the *ForeView* clients: *fvtopomap*, etc.) via *ForeView*'s database *fvdb*. After the device information is retrieved, the *fvdisco* NIAM stores network topology information in the *ForeView* database. *ForeView* applications then insert the appropriate information into the OpenView database and the devices are displayed in the OpenView ATM Networks submap.

#### 4.1.1.3 Seeding the Discovery

To perform the discovery, you must specify one or more *seed switches* when *ForeView* is first started. From this initial ATM switch, *ForeView* will go on to discover all ATM devices and links connected to this switch and subsequent devices. You must specify at least one ATM switch from each ATM cloud to be discovered (an ATM cloud being any contiguous or connected set of ATM switches). To assist *ForeView* in discovering large networks, you specify multiple seed switches and use the *fvseed* command, described in the following section.

The seeds for network discovery can be provided during the initial installation of *ForeView*, or by using the *fvseed* command line option.

#### 4.1.1.4 Fyseed Command-Line Arguments

The following are command-line arguments for seeding network discovery:

**fvseed** [nameorlPAddr...] The list of switches given to fvseed must be a commaseparated list of switch names or IP addresses.

The standard *ForeView* command flags *–help*, *-tracingOn*, and *–traceFile* also are accepted.

#### 4.1.1.5 Fvdisco Command-Line Arguments

The main parameter that you can control is the <code>tenacity</code>, which controls how many system and network resources *fvdisco* devotes to discovering the network. *Fvdisco* reasonably distributes the CPU and bandwidth resources among different priorities according to this parameter.

The following are command-line arguments for network discovery:

**fvdisco -discint**  *seconds* Specifies the discovery interval, where *seconds* is the

interval in seconds between subsequent discovery

cycles. The default is 120 seconds.

**fvdisco -tenacity > controls** how fast *fvdisco* should perform discovery.

A value greater than zero and less than or equal to

1.0, 1.0 being the fastest.

The standard ForeView command flags – help, -tracingOn, and -traceFile also are accepted.

## 4.2 The Monitoring NIAM (fvmon)

The Status Monitoring NIAM (*fvmon*) monitors the status of discovered nodes and links in the network and makes this information available to its clients (*fvtopomap*, etc.) via *ForeView*'s *fvdb* database.

#### 4.2.1 What Fymon Does

The *fvmon* NIAM is responsible for polling SNMP agents residing on ATM switches to determine any status changes to discovered nodes and links in the network. A node can be one of the following:

- A single ATM switch fabric
- A host
- A LAN access device such as an ES-3810 or PowerHub

A link is normally associated with two nodes (which can be two ATM switches or one ATM switch connected to an access device). For each link, *ForeView* provides three status types:

- Hardware carrier status
- · UNI signalling status
- SPANS signalling status

Most links will have UNI and SPANS signalling by default. By providing a separate status for each, you can have a finer understanding of the status of a link. In some cases, a link may not have signalling at all and hardware carrier status will indicate the link working status.

### 4.2.2 How Fvmon Works

The *fvmon* NIAM functions in the following capacities:

- At start-up, the *fvmon* NIAM establishes a connection to *fvdb* and retrieves the managed entities (nodes and links) for monitoring.
- The ATM switches are polled routinely to determine any status changes to discovered nodes and links in the network.



The *fvmon* command-line argument -tenacity controls the rate at which *fvmon* NIAM updates the managed entities and is explained in more detail later in this chapter.

Records changes in status for the monitored entities in the database, creating a
running history of status changes to date. The *ForeView* database maintains status
tables for nodes, hardware carrier links, UNI signalling links, and SPANS signalling links. Status changes are time-stamped and stored in chronological order.

#### 4.2.2.1 How Fymon Determines the Status of Nodes

As previously stated, a node can be a single ATM switch fabric, a host, or a LAN access switch. For ATM switches, the node status is determined by SNMP reachability via any of the known IP addresses in the database. The status is determined by the following:

• The sysObjectId of a switch is queried. If successful, the status of the node is marked UP. If unsuccessful, the status of the node is marked DOWN.



The *fvmon* NIAM checks all of the known addresses in the database before concluding that an ATM switch node is DOWN. In addition, the "primary address" of the node will be updated to the last UP address. However, if none of the addresses are UP then the first known address will be set up as the primary address of the node.

- If the status of an ATM switch node is determined to be DOWN, then all the links originating or terminating on the node are inferred to be DOWN.
- If during the course of determining a link status, an associated node is determined to be unreachable (not responsive to SNMP queries), that node is marked DOWN. Conversely, if an associated node is found to be reachable, that node is marked UP.

For ATM-connected LAN access devices (ES-3810s, PowerHubs, etc.), the status is determined by way of link status:

- The link status between a device and its neighboring ATM switch is examined, and if there is at least one link (hardware carrier or signalling) UP, it can be inferred the device must be UP (there can be more than one link between devices).
- If all links between a device and its neighboring ATM switch are examined and found to be DOWN, it can be inferred the device also must be DOWN.

#### 4.2.2.2 How Fymon Determines the Status of Links

As previously stated, there are three status types associated with links:

- Hardware carrier status as derived from the MIB variable hwPortOperStatus
- UNI signalling status as derived from the MIB variable q2931ILMIOperStatus
- SPANs signalling status as derived from the MIB variable sigPathOperStatus

A link normally is associated with two nodes (which can be two ATM switches or one ATM switch connected to an access device). Therefore, link status can be derived based on the MIB variable status reported by either end of a link. A link can be UP, DOWN, UNKNOWN or NON-EXISTENT. An endpoint used to evaluate link status must meet the following criteria:

- The endpoint must be a ATM switch.
- The endpoint node must have at least one IP address associated with it.
- The endpoint node must be "managed".

The following table summarizes the status values associated with each link type.

Type of Link **Status Conditions** Hardware Carrier **DOWN** Hardware carrier is DOWN, or, The ATM switch did not respond to SNMP on any of the known IP addresses and is marked DOWN. UP The Hardware carrier is UP, and, The ATM switch queried is also UP. **UNKNOWN** Could not find a valid endpoint node (ATM switch) to query. **NON-EXISTENT** n/a

**Table 4.1 - Link Status Summary** 

4 - 8

**Table 4.1 - Link Status Summary** 

Type of Link	Status	Conditions
UNI Signalling	DOWN	UNI signalling is down, or,
		Hardware carrier is down, or,
		The ATM switch did not respond to SNMP on any of the known IP addresses and is marked DOWN.
	UP	The UNI signalling is UP.
	UNKOWN	Could not find a valid endpoint node (ATM switch) to query.
	NON EXISTENT	The UNI signalling does not exist. This can happen if it is a hardware carrier only or SPANS only link.
SPANS Signalling	DOWN	SPANS signalling is down, or,
		Hardware carrier is down, or,
		The ATM switch did not respond to SNMP on any of the known IP addresses and is marked DOWN.
	UP	The SPANS signalling is UP.
	UNKNOWN	Could not find a valid endpoint node (ATM switch) to query.
	NON EXISTENT	The SPANS signalling does not exist. This can happen if it is a hardware carrier only or UNI only link.

#### 4.2.2.3 How *Fymon* Adapts to Network Conditions

The *fvmon* NIAM is able to adapt to changes in the network. For example, if a change in status is detected in one of the entities being monitored, *fvmon* might update the entity under consideration more frequently. If *fvmon* does not notice any more changes in the status of the object, the frequency is gradually reduced back to the default.

#### 4.2.2.4 How Fvmon Implements Tenacity

Monitoring network status may overutilize CPU resources as well bandwidth for SNMP traffic. Because of this, the "tenacity" feature allows the network administrator to control the rate at which the *fvmon* NIAM works. The range of values for tenacity is greater than zero and less than or equal to 1.0. A tenacity of 1.0 asks *fvmon* to work as fast as possible. By default the tenacity value is set to 0.5. When the tenacity is 0.5, it is expected that the *fvmon* NIAM work only 50% of the time.



The tenacity value can be specified <u>only</u> at startup time.

#### 4.2.2.5 Status Records

As *fvmon* monitors the network, it adds entries to the database tables. It never changes or deletes entries, so that the *ForeView* database contains a complete record of status changes for the nodes and links in the database.

These status entries will cause the size of the ForeView database to continually grow. You can run *fvdbtrim* to remove all but the current status line from the status tables, as described in "Removing Old Status Entries from the Database" on page A-7.

Refer to Appendix B for more information about the *ForeView* database.

#### 4.2.2.6 Fymon Command-Line Arguments

The main parameter that you can control is the tenacity, which is the rate at which *fvmon* attempts to monitor the network.

The following are command-line arguments for status monitoring:

fvmon -tenacity < tenacity>

Controls how fast *fvmon* should poll. Must be greater than zero and less than or equal to 1.0, 1.0 being the fastest.

The standard ForeView command flags -help, -tracingOn, and -traceFile also are accepted.

## 4.3 How *ForeView* Stores Database Information

*ForeView* uses a relational database to store the network topology information. This database is populated by the *fvdisco* NIAM and the status information in the database is updated by the *fvmon* NIAM. The *ForeView* applications also interact with the database:

- The *fvtrapd* application updates the database with information from trap events, such as a change of status.
- The *fvovmon* application updates the database when an administrator unmanages or manages a switch via the *ForeView* maps.

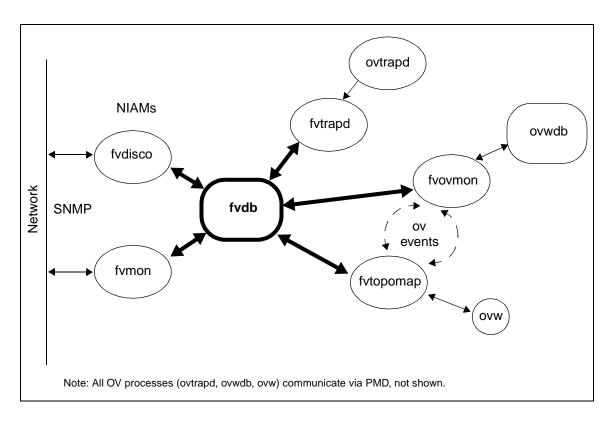


Figure 4.2 - The ForeView Database Stores Node, Link, and Status Information

#### 4.3.1 The Database Schema

The *ForeView* database schema reflects the topology of an ATM network. Tables exist for each of the important elements of the network. Relations between table entries organize the information to show the topology of the network. The database tables are described in Appendix B.

The database includes the following discovery tables. These tables are populated when the Discovery NIAM discovers the network.

- Nodes
- Links
- Addresses
- Interfaces
- Containers
- Endpoints
- Signalling entities
- Signalling VPIs.

The *ForeView* database also includes the following status tables. These tables are populated and updated as the Monitor NIAM monitors the network.

- Node status
- Hardware carrier status
- UNI link status
- SPANS link status

In addition to the topology and status information, the database also includes tables used for management of change notifications, and other management information.

## 4.3.2 Adding Non-FORE Devices to the Database

*ForeView* discovers managed FORE ATM switches, ATM-connected FORE access devices, and the associated NNI and UNI links only (see page 5-14 for information on managing and unmanaging objects in *ForeView*). Because of this, any non-FORE ATM switches or UNI devices at the far end of a link originating from a FORE ATM switch cannot be discovered or mapped.

The *fvaddlink* utility allows users to specify these non-FORE ATM switches and UNI devices and to have those devices mapped in the OpenView ATM Networks submap and monitored by the *fvmon* NIAM.

#### 4.3.2.1 Fvaddlink Command-Line Arguments

Use *fvaddlink* to describe a link between a FORE ATM switch and a non-FORE ATM switch or UNI device. The link specifications are added into the *ForeView* database. The following are command-line arguments for adding undiscovered links:

fvaddlink -switch <switchnameorip></switchnameorip>	The DNS name or IP address of a FORE ATM switch that has been discovered and has an entry in the <i>ForeView</i> database.
fvaddlink -port <portid></portid>	The name (in BNP format) or port index of a port on the FORE ATM switch that is the source of the link.
fvaddlink -remote <remotedevnameorip></remotedevnameorip>	The DNS name or IP address of a non-FORE ATM switch or UNI device that is at the far end of the specified link.

The standard ForeView command flags -help, -tracingOn, and -traceFile also are accepted.

Fvaddlink will fail in the following cases:

- The specified switch is an unknown DNS name or invalid IP address, is an address that does not appear in the *ForeView* database, or is not the address of a FORE ATM switch.
- The specified port ID is not in valid BNP or port index form, or is not valid for the specified switch.
- The remote device specified is an unknown DNS name or invalid IP address, or it identifies a device which is a FORE ATM switch.
- A link already exists on the specified port and switch.
- No connection with the ForeView database could be established, the connection
  was lost, or read/write failure occurs when accessing the database.

#### 4.3.2.2 Running Fvaddlink from the OpenView Maps

Fvaddlink also can be run in GUI mode from the OpenView submaps via the menu option ForeView: Add Link. If a switch on the OpenView map is selected, and ForeView: Add Link is selected, the fvaddlink dialog pops-up, with the IP address of the selected switch filled in automatically. At some time after successful link creation, and assuming fvmon and fvovmon are running, the new link and remote device will appear on the ATM Connections submap for the specified ATM switch.



If any of the *-switch*, *-port*, and/or *-remote* flags are omitted from the command line invocation of *fvaddlink*, it will go into GUI mode. Each argument provided on the command line will be automatically inserted into the appropriate text field in the dialog.

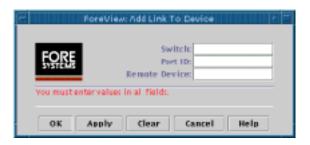


Figure 4.3 - The Fvaddlink Dialog

The text area in the middle of the dialog is for display of status (normal font) and error (red font) messages. The buttons cause the following actions, respectively:

**OK** Attempts to create a link from the information specified in the text boxes. If successful, the dialog is dismissed; otherwise, an error message is displayed.

Apply Attempts to create a link from the information specified in the text boxes. If successful a success status message is displayed and the OK and apply buttons are disabled. (The buttons are reactivated if any changes are made in the text boxes.) Unsuccessful attempts display an error message.

**Clear** Clears all text boxes.

Help

**Cancel** Dismisses the dialog immediately, without attempting to create a link from the information displayed in the text boxes.

Displays Help for this dialog.

4 - 15

## 4.4 How ForeView Integrates with HP OpenView

*ForeView* uses three background processes to integrate with HP OpenView and ensure that an accurate and current representation of the ATM network appears in the OpenView maps.

## 4.4.1 The ForeView Applications

#### 4.4.1.1 The Fvtopomap Application

The OpenView mapping application *fvtopomap* creates an OpenView submap and populates it with symbols to represent the near-physical topology of an ATM network. It takes information from the *ForeView* database and places it on the map. This includes status information.

#### 4.4.1.2 The Fvovmon Application

The application *fvovmon* is responsible for keeping the *ForeView* database (*fvdb*) synchronized with the OpenView database (*ovwdb*). For every object in *fvdb*, *fvovmon* creates an object in *ovwdb*. In addition, it interacts with *fvtopomap* and the NIAMs (via *fvdb*) to perform the following tasks:

- Generate events for *fvtopomap* when new objects are added to *ovwdb* or objects in *ovwdb* are modified
- Handle events from *fvtopomap* (such as those generated through user interactions like Manage or Unmanage)
- Generate events for the discovery NIAMs (event passing done through *fvdb*) for events generated through user interactions

#### 4.4.1.3 The Fvtrapd Application

The application *fvtrapd* is responsible for monitoring FORE ATM-specific traps sent to the management station. It updates node and link statuses in the *ForeView* database depending on the trap information.

## 4.4.2 The Topology Mapping Process

The application *fvtopomap* is an OpenView mapping application that creates an OpenView submap and populates it with symbols to represent the near-physical topology of an ATM network, including ASX switches, NNI, and UNI links, and connected ATM devices.



The near-physical representation of an ATM network topology means that links, in general, represent physical links in the ATM network. However, in the case where VP tunneling is used, the links represent logical links between devices.

The features of *fvtopomap* include the following:

- FORE Systems' networked devices are represented according to the functional abilities of the devices.
- Multi-agent devices are represented using a single symbol, further highlighted with a submap containing the individual fabrics.
- Status updates of managed objects, including links. Changes in status may be the
  result of events received by *fvtopomap* in response to SNMP traps sent by the
  devices.
- Support for the creation of "location" symbols and submaps, including moving symbols in and out of them (except for location symbols used for representing an enclosure).
- The ability to "manage" or "unmanage" a device via the user interface (refer to "Note About the Unmanage Objects Command" on page 5-14).

This application is a well-behaved OpenView process managed by *ovw.* See HP OpenView's *Using Network Node Manager* documentation for more information.

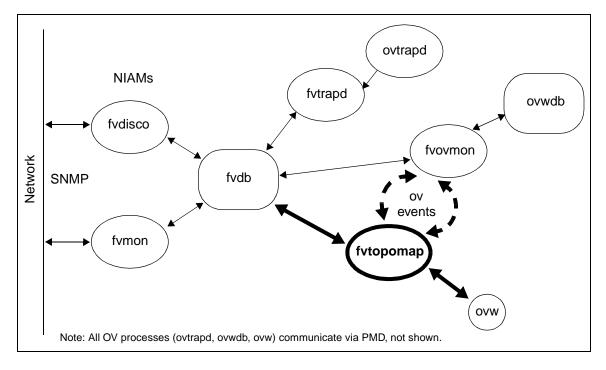


Figure 4.4 - Fvtopomap Creates the OpenView ATM Map

#### 4.4.2.1 Fvtopomap Arguments

The following are the arguments for *fvtopomap*:

fvtopomap -totalSyncInterval <seconds>

Specifies the interval for total synchronization, where *seconds* is the interval in seconds between total synchronizations. The default is 21600 seconds (six hours).

fvtopomap -overrideUserChanges

Turns on overriding user changes. When this parameter is used, when *fvtopomap* updates the *ForeView* map, it will override manual changes that users have made to the map.

The standard ForeView command flags -help, -tracingOn, and -traceFile also are accepted.



*Fvtopomap* cannot be run directly from the command line, therefore arguments need to be specified in the *fvtopomap* registration file.

## 4.4.3 The Database Synchronization Process

The application *fvovmon* is responsible for keeping the *ForeView* database (*fvdb*) synchronized with the OpenView database (*ovwdb*).

When *fvovmon* first starts up, it does its initialization and then performs a total synchronization in which *fvovmon* retrieves all the records from *fvdb* and synchronizes them with records in *ovwdb*. After that, *fvovmon* starts a timer to perform regular synchronizations, and also waits for events from *fvtopomap*, such as user interactions and OVW events (like ovstop).

The following summarizes the event flow that happens when a symbol is added to a map:

- 1. *fvtopomap* sends a message to *fvovmon* about the event.
- 2. *fvovmon*, upon receiving the event, informs the NIAM about the user action by creating a table entry in the notification table.
- 3. The NIAM then goes to the network and finds out about the added symbol.
- 4. *fvovmon*, upon detecting some new object in *fvdb*, creates a corresponding object in the *ovwdb* and then sends an event to *fvtopomap*.
- 5. *fvovmon* registers the *ForeView* workstation in the trap destination list of managed switches, described in "The Trap Destination Table" on page 4-22.

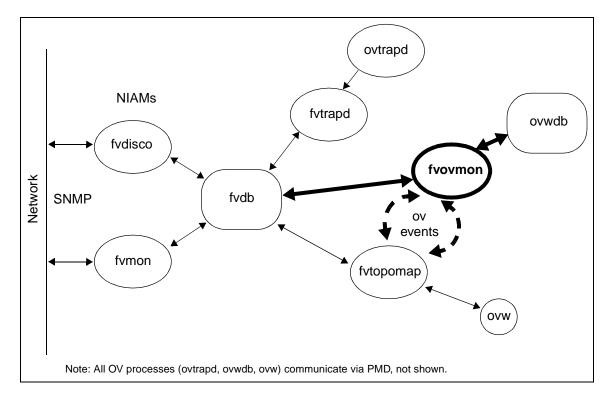


Figure 4.5 - Fvovmon Synchronizes ovwdb with fvdb

#### 4.4.3.1 Fvovmon Command-Line Arguments

The following are command-line arguments for network monitoring:

fvovmon -totalSyncTime <seconds> Specifies the interval for total synchronization, where *seconds* is the interval in seconds between total

synchronizations. The default is 3600 seconds.

fvovmon -changesPollInterval

<seconds>

Specifies the interval for routine polling, where *seconds* is the interval in seconds between regular synchronization polls. The default is 15 seconds.

**fvovmon -trapRegistrationOff** Deactivates trap registration.

fvovmon -overrideUserChanges Turns on overriding user changes. When this parameter is used, when *fvovmon* updates the

Fore View map, it will override manual changes that

users have made to the map.

The standard command flags -help, -tracingOn, and -traceFile also are accepted.



Default values are stored in *fvdb*. Command line options override *fvdb* values at start-up time.



Arguments for *fvovmon* typically are specified in its associated LRF file, not from the command line.

This application is a well-behaved OpenView daemon managed by the *ovspmd* process. See HP OpenView's *Using Network Node Manager* documentation for more information.

## 4.4.4 The Trap Handling Process

The application *fvtrapd* is responsible for receiving and handling FORE ATM switch traps forwarded by OpenView's *trapd*. Status change information contained in a forwarded trap is recorded in the *ForeView* database (*fvdb*). The creation of a status change record triggers notification to the Status Monitoring NIAM, which in turn performs additional status verification. Changes in status are propagated by *fvovmon* to *fvtopomap*, which updates the ATM submap display.

For example, if a switch detects that an inter-switch connection becomes non-operational, the switch sends an asxLinkDown trap to the network management station. The *fvtrapd* application receives the trap, checks that it is a *ForeRunner* switch trap, decodes the trap message, checks the database, and updates the link status in the database.

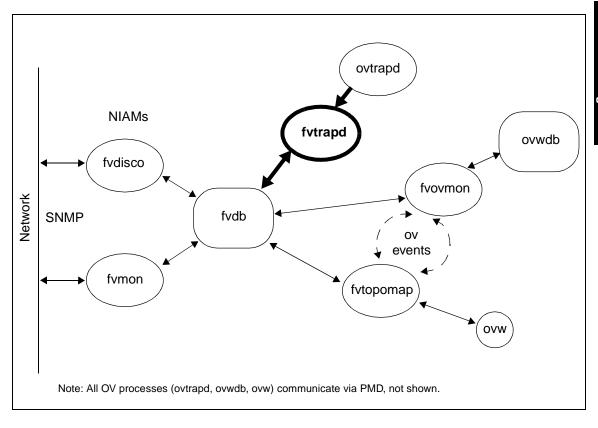
#### 4.4.4.1 The Trap Destination Table

All *ForeRunner* switches have a table of trap destinations. When the switch detects a change in the operational status of one of its ports, it sends a trap to all network management stations that appear in the trap destination list. This list is represented by a MIB table, and can be written using SNMP set messages. When *fvovmon* detects a new switch in the network and polls it, it sets an entry in the switch's trap destination table for the IP address of the machine on which *fvovmon* is running. From this point on, *ForeView* receives traps from this switch, and changes in the status and topology of the network are reflected in the OpenView maps almost immediately.

#### 4.4.4.2 Supported Traps

Fvtrapd supports the following traps:

- warmstart
- coldstart
- asxlinkup
- asxlinkdown
- asxspansup
- asxspansdown
- asxq2931up
- asxq2931down



**Figure 4.6 -** *Fvtrapd* Handles ATM Switch Traps

## 4.4.4.3 Fvtrapd Arguments

The standard ForeView command flags -help, -tracingOn, and -traceFile are accepted for fvtrapd.



Arguments for *fvtrapd* typically are specified in its associated LRF file, not from the command line.

This application is a well-behaved OpenView daemon managed by the *ovspmd* process. See HP OpenView's *Using Network Node Manager* documentation for more information.

#### 4.4.4.4 Trap Configuration in HP OpenView

Traps can be viewed in the OpenView Event Categories window. This window has six predefined categories, each with a corresponding button. A change in button color indicates that an event occurred for that category and a corresponding trap has been sent.



When you click on a button in the Event Categories window, a window listing the events appears.

The six predefined categories are as follows:

**Error Events** Indicates an inconsistent or unexpected behavior has

occurred.

**Threshold Events** Indicates a threshold has been exceeded.

**Status Events** Indicates an object or interface status has changed to

up or down, or an object or interface has started or

stopped responding to ICMP echo requests.

**Configuration Events** Indicates a change to the configuration of a node.

**Application Error Events** Indicates an HP OpenView application generated an

alarm or alert.

All Events Includes all of the above events and any others in

one dialog.

As previously stated, a change in button color indicates that an event occurred. The color of the button indicates the severity of the event. HP OpenView allows you to change the severity of a trap and to create new trap categories. For example, you can change the trap severity of link status (up/down) from Warning to Major.

#### 4.4.4.4.1 Changing Trap Severity

Use the HP OpenView console to change trap severity. For example, the following steps explain how to change the trap severity of Link Down from Warning to Major.

- 1. Pull down the Options menu item and click on Event Configuration.
- 2. Select atmswitch to view FORE-specific traps.



Selecting **FORE\_ASX** will produce:

- a list of traps (Event Name)
- the Sources.
- 3. Highlight fore asxlinkdown.
- 4. Select on the Edit->Modify option to bring up the Modify Event dialog.
- 5. Change the Severity to Major (pull down the Severity menu and click on the selection).



As an option, you can create a custom pop-up notification or an automatic command response for the trap from this dialog.

- 6. Click **OK** to return to the Event Configuration dialog.
- 7. To end this process, Click **OK** to close the Event Configuration dialog.

#### 4.4.4.4.2 Creating a Trap Category

Software messages generated by *ForeView* appear in the Configuration Events and Status Events categories. You can create a new trap classification to log these *ForeView*-generated messages, creating a *ForeView*-specific category for ease of viewing. These traps still will be included in the All Events category.

The following steps explain how to create a new trap category.

- 1. Pull down the Options menu item and click on Event Configuration.
- 2. Select OpenView from the Enterprise Identification list.
- 3. Highlight FV\_Message in the Event Identification window.
- Select the Edit->Configure->Event Categories option to launch the Configure Event Categories dialog.



This dialog lists the items seen in the Event Categories window.

- 5. Enter a new category name (FV Events, for example).
- 6. Click the **Add** button to complete this action.
- 7. Click on the Close button.
- 8. Highlight FV\_Message in the Event Identification window.
- 9. Click on the **Modify** button.
- 10. Pull down the Event Category menu item and click on the new category just created (FV Events, for example).
- 11. Click on the OK button.
- 12. Click on the OK button on the Event Configuration dialog.

# 4.5 Changing Community Strings

If you change the community strings for devices in your network, you must:

- 1. Change the community string for the device as you normally would in OpenView. You can either:
  - Use the SNMP Configuration option on the Option menu.
  - Use the xnmsnmpconf utility from the command line.
- 2. Run the *fvovsnmpconf* utility to update the *ForeView* database with the new community strings.
- 3. Restart the ForeView NIAMs.

The standard *ForeView* command flags *-help*, *-tracingOn*, and *-traceFile* are accepted for the *fvovsnmpconf* utility.

## 4.6 Summary of *ForeView* Commands

Table 4.2 lists the *ForeView* commands and the corresponding command-line arguments. Note that all commands include the following standard arguments. These arguments are not shown in the table:

-help Displays usage information.

**-tracingOn** Turns on tracing.

For UNIX, tracing output defaults to the file /usr/fore/foreview/tmp/command-name.trc.

For Windows NT, tracing defaults to the file

\foreview\tmp\command-name.trc

NOTE: The tracing files may quickly become very

large.

-traceFile <traceFileName> Specify a filename and directory for the trace file

other than the default.

**Table 4.2 - Summary of** *ForeView* Commands

Command	Arguments	Purpose	Refer to
fvstart	[fvdisco fvmon]	Starts <i>ForeView</i> NIAMs (UNIX-only).	page 3-3
fvstop	[fvdisco fvmon]	Stops ForeView NIAMs (UNIX-only).	page 3-3
fvsrvc	-start [fvdisco fvmon] -stop [fvdisco fvmon] -status [fvdisco fvmon]	Starts, stops, or gives the status of the <i>ForeView</i> processes. (Windows NT-only).	page 3-3
fvstatus	[fvdisco fvmon]	Displays the status of the Fore- View processes. (UNIX-only)	page 3-3
fvseed	[nameorIPAddr]	Specifies seed switches for <i>Fore-View</i> discovery.	page 4-4
fvtopomap	-totalSyncInterval <seconds> -overrideUserChanges</seconds>	Creates OpenView map and populates with symbols representing the ATM network.	page 4-18
fvaddlink	-switch <switchnameorip> -port <portid> -remote <remotedevnameorip></remotedevnameorip></portid></switchnameorip>	Adds non-FORE ATM switches and UNI devices to the database.	page 4-13
fvmon	-tenacity <tenacity></tenacity>	Updates information in <i>fvdb</i> with information from network.	page 4-10
fvovmon	-totalSynchTime -changesPollInterval <seconds> -trapRegistrationOff -overrideUserChanges</seconds>	Synchronizes <i>fvdb</i> with the OpenView database.	page 4-10
fvtrapd		Receives and handles ATM switch traps.	page 4-22
fvovsnmpconf		Ensures consistent configuration of SNMP community strings.	page 4-27
fvovdbclean		Removes ForeView symbols from the OpenView map.	page A-5
fvovdbtrim		Removes all but latest status entries from <i>fvdb</i> .	page A-7

Network Discovery and Monitoring

# CHAPTER 5

# **Network Mapping**

This chapter provides information on how to map a FORE Systems ATM network using *Fore-View* under HP OpenView. *Fvtopomap* is an OpenView mapping application that creates an OpenView map and populates it with symbols to represent the near-physical topology of an ATM network. This enables *ForeView* to provide:

- Mapping of near-physical topology of the discovered ATM network, including ASX switches. NNI. and UNI links. and connected ATM devices.
- Representation of FORE Systems' networked devices according to functional abilities of the devices.
- Representing of each multi-agent device using a single symbol with a submap containing the individual fabrics.
- Status updates of managed objects, including links. The status changes may be
  the result of an event received by *fvtopomap* in response to an SNMP trap sent by
  the device itself.
- Support for the creation of location symbols and submaps, including moving symbols in and out of them (except for location symbols used for representing an enclosure).
- Support for the Manage and Unmanage features from the user interface.
- Copying connected ATM device symbols to the ATM Network submap.



This chapter includes screen examples showing OpenView running on UNIX. When OpenView is running on Windows NT, there are slight differences in the appearance of the OpenView windows.

## 5.1 The ForeView ATM Maps

*ForeView's fvtopomap* is an application that is integrated under HP OpenView Windows. *fvto-pomap* reads ATM topology data and displays it in a series of submaps.

- Root level submap
- ATM level submap
- Switch connections submap
- Redundant NNI links submap.

In addition to viewing the information on the ForeView submaps, you can manipulate the maps using standard OpenView commands.

### 5.1.1 Physical vs. Logical Topology

The *ForeView* maps reflect the logical topology of the ATM network. In general, the links shown represent physical links in the ATM network. However, when Virtual Path (VP) tunneling is used, the links represent logical links. The links do not reflect the devices that the VP "tunnels" through.

Other than the case of VP tunneling, the physical and logical links are identical.

## 5.1.2 Root Level Submap

In the **Root Level** submap, a symbol for the entire ATM network is presented together with the IP internet symbol. The color of the symbol represents the status of the ATM network.

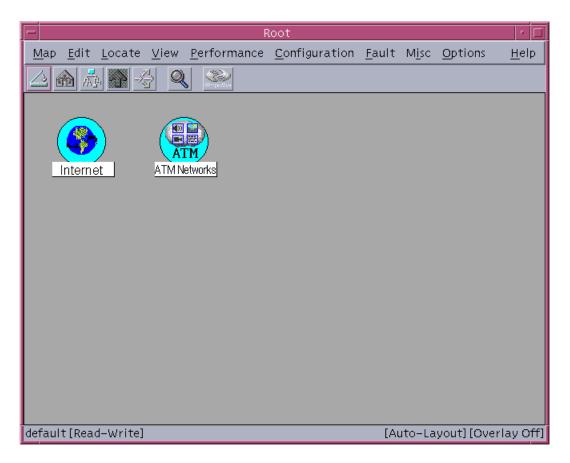


Figure 5.1 - Level 1 Submap - Root Level

## 5.1.3 ATM Networks Submap

Double-clicking on the ATM Networks symbol launches the **ATM Networks** submap. This submap displays all the switches and the interswitch connections that exist in the managed network.

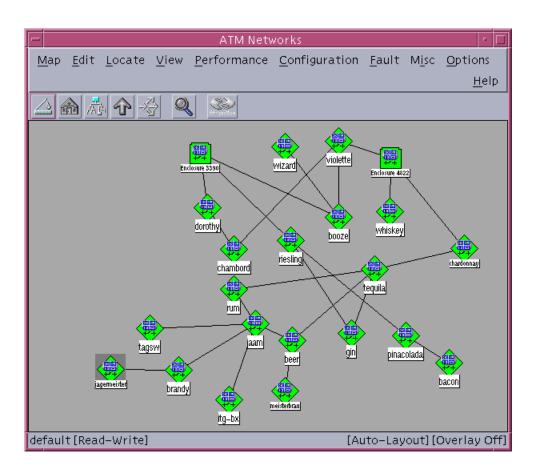


Figure 5.2 - Level 2 Submap - ATM Networks

## 5.1.4 Switch Connections Submap

Double-clicking on a switch symbol brings-up the **Switch Connections** submap. In this submap, the switch is displayed in the center of the screen and all the attached devices (hosts, switches, ATM-access) are displayed around it in a star layout, with the port ID shown for each link.

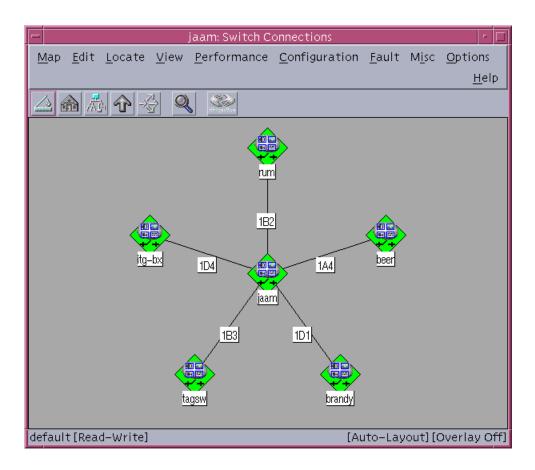


Figure 5.3 - Level 3 Submap - Switch Connections

### 5.1.5 Fabrics Submap

Double-clicking on an ASX-1000 switch symbol brings up the **Switch Fabrics** submap. An ASX-1000 can be populated with up to four fabrics. This submap displays the valid switch fabrics in the enclosure.

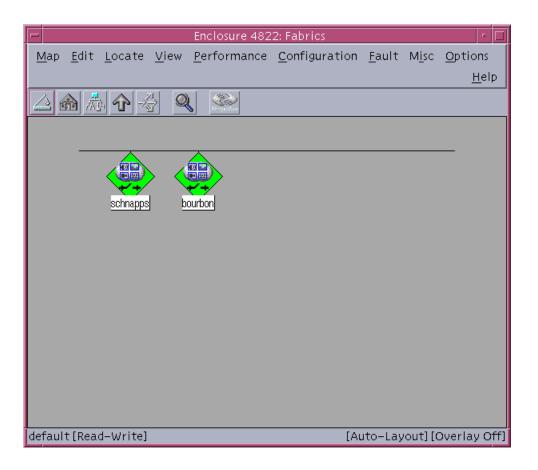


Figure 5.4 - Switch Fabrics Submap

5 - 6

## 5.1.6 Redundant NNI Links Submap

A line between switch icons in the ATM Network Submap indicates a bidirectional connection between those switches. If there are redundant NNI links between the switches, you can double click on the line to bring up a submap with all links between the two switches. Each link label is composed by the two ends of the physical link, <code>switchl-name:np</code> <--> <code>switchl-name:np</code>. The NP notation is derived from the network module identifier (A-D) plus the actual port number.

## 5.2 Launching the ATM Switch Element Manager

This release of *ForeView* provides access to a new method of ATM switch configuration and management through an easy-to-use World Wide Web (WWW) interface using a Web browser such as Netscape or Microsoft's Internet Explorer. The WWW provides a graphical interface with a simple structure that makes use of internet presentation attributes common in the HTML (Hypertext Markup Language) environment such as pages, frames, and tables.

The WWW interface, called an Element Manager, mirrors the command line user interface known as the ATM Management Interface (AMI). To launch an Element Manager against a FORE Systems ATM switch, select a switch icon from either the **ATM Networks** submap or the **Switch Connections** submap, and select the Misc|Web Browser|Management Page option from the OpenView menu.

Enter the IP address of your switch in the URL Location field of the browser to access the Element Manager. At the login screen (shown in Figure 5.5), you must enter your Username. On a new switch running *ForeThought* 5.3.x, enter the default username as ami. This username is assigned a null password.

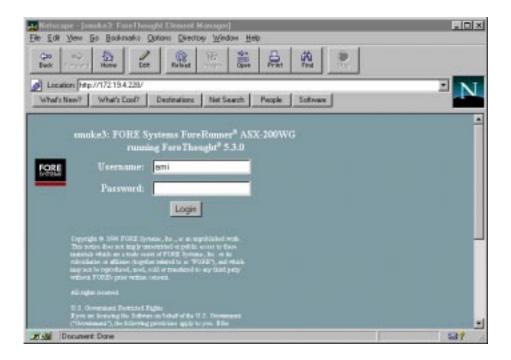
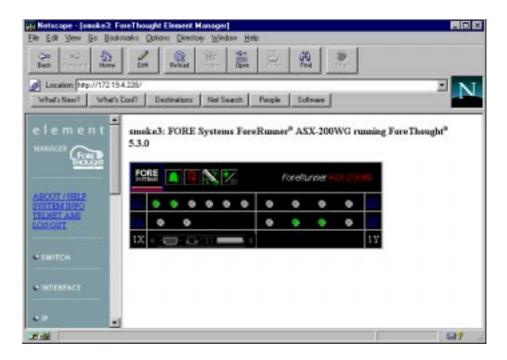


Figure 5.5 - Element Manager Login Screen

The Element Manager provides a graphical interface to many of the AMI commands that allow you to configure and query various hardware and software aspects of switches and network modules. A graphical representation of an actual FORE Systems' ATM switch, including the number and type of network modules installed in the device and the status of the ports on each of these modules, is provided when you login to a switch, as shown in Figure 5.6.



**Figure 5.6 -** Element Manager's ATM Switch Front Panel

#### 5.3 How Status Colors are Derived

The color of the ATM object (switch, link, or ATM-access device) represents its status. When a symbol becomes red, that means that the device can not be reached via SNMP from the management station.

If the color of a switch is not green, it indicates that the switch (or one or more fabrics in the switch) is down.

The color of the ATM networks symbol in the **Root Level** submap represents the status of the entire ATM network. The symbol is blue if there are no ATM objects in its submaps. It is green if all switches and interswitch links in the map are OK. It becomes yellow, cyan, orange, or red if there are some ATM switches or interswitch links that are down.

For a description of how the *ForeView* NIAMs determine the status of devices, refer to "How Fvmon Determines the Status of Nodes" on page 4-7.



The status of ATM switches and edge devices in the network is unaffected by link failures to endhosts. From the ATM network perspective, the failure of a switch or an interswitch link is much more important than the failure of a host or a link to the host. Therefore, the status of a host does not propagate to the top level ATM Networks symbol.

Interswitch connections are also displayed on the **Switch Connections** submap. All links in this submap are labeled by the switch's port label to which they are connected. The color of the link indicates the signalling status. The color of the remote node indicates whether the node can be reached (via SNMP) from the management station. However, if the remote node is a UNI device, its status is derived from the link status.



The *unknown* link status appears the same as an *up* link status, represented as a black line.

## 5.4 Managing Symbols and Maps

ForeView 5.0 allows you to manipulate symbols in the ForeView maps as you would in the OpenView IP map.

### 5.4.1 Creating Location Symbols

*ForeView* running under OpenView supports a feature that allows you to create a "location" symbol that represents multiple symbols in a single location in the managed network. When created, other symbols from a submap then can be cut and pasted in a related submap launched from the "location" symbol. This submap is referred to as the "location submap".

The status of the location symbol is derived from the combined status of the symbols in a location submap. The use of the location symbol increases the usability of OpenView in a large network because multiple symbols are represented by a single symbol on the original submap and the complete network representation is partitioned into multiple submaps.



Symbols can be created from the "Location" symbol class only, on the "ATM Networks" submap only.



You must not use the "Enclosure" location symbol as a general location symbol. The *fvtopomap* application uses the "Enclosure" location symbol to represent multiple switch fabrics in an enclosure (e.g., an ASX-1000).

### 5.4.2 How Location Symbols are Implemented

The following are requirements for the successful implementation of location symbols:

- The creation of location symbols, and the automatic creation of the location submap, is supported in the "ATM Networks" submap and in the location submaps only.
- Symbols from a submap are cut-and-pasted or dragged-and-dropped (in UNIX) onto a location submap.
- Symbols from a location submap can be cut-and-pasted back to a regular submap or to another location submap.
- The status of links to and from symbols moved from one submap to another are updated using the model implemented by HP OpenView's ipmap process in the "IP Networks" submap hierarchy.
- Assignment of symbols to location submaps is maintained when ovw is restarted.



If a symbol is cut from one submap, it must be pasted onto another. If the symbol is not pasted in a timely manner (i.e., before fvtopomap starts a synchronization cycle), then fvtopomap will paste the symbol onto the "ATM Networks" submap.



This feature does not support the dynamic creation of new symbols. Only symbols that have previously been cut from a submap can be added to another submap.

# 5.5 Retained Links When Pasting Edge Devices

You can copy and paste ATM edge devices (*PowerHubs* and ES-3810s) to the ATM Networks submap or a newly created location submap. All links will be mapped correctly, even if the device is dual or multi-homed.



Be sure to copy the edge device symbols. Do not cut them from their original maps.

## 5.6 Note About the Unmanage Objects Command

Note that to unmanage an object on an ATM map in OpenView you must use the Fore-View: Unmanage Objects command under the Options menu, **not** the command under the Map menu. This command (when executed under the Options menu) will ensure that the selected device is no longer discovered or monitored by *ForeView*.

Likewise, you can use the ForeView: Manage Objects command under the Options menu (not the Manage Objects command under the Map menu) to resume monitoring of the device.

#### CAUTION



If you use the standard OpenView Unmanage command, found under the Map menu, you may unmanage the entire network.

Use the ForeView: Unmanage Objects command under the Options menu.

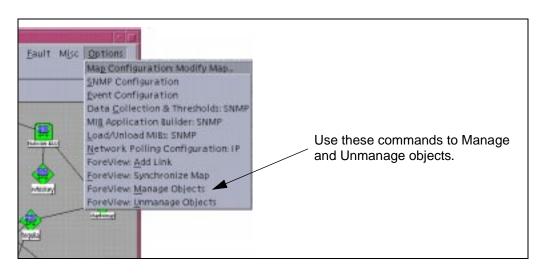


Figure 5.7 - ForeView: Manage Objects Command



# APPENDIX A Tips and Troubleshooting

## A.1 Checking ForeView Processes

As with all other HP OpenView background processes, fvovmon and fvtrapd should be running whenever ForeView is running. Normally, the background processes are started by running ovstart. To verify that processes are running, type:

#### ovstatus

The result shows you the status of all OVW daemons that are supposed to be running, and their status. You should see the following for fvovmon and a similar entry for fvtrapd:

> object manager name: fvovmon

behavior: OVs\_WELL\_BEHAVED

state: RUNNING PID: <pid>

last message: Initialization complete.

exit status:

If ovstatus indicates that there is a problem with fvovmon or fvtrapd, try to restart the process by typing the following as root:

ovstart fvovmon

or:

ovstart fvtrapd

### A.1.1 Checking Fvtopomap

Checking the status of *fvtopomap* is different because it is launched when the **ovw** command is executed. The following procedures explain how to check *fvtopomap* on both UNIX and Windows NT.

#### A.1.1.1 Checking Fvtopomap on UNIX

With OpenView running, find the FORE JRE process with the parent process ID. This process is *fvtopomap*. Use the following procedure to determine if *fvtopomap* is running:

1. Run the command:

It will show several lines of output.

- 2. Note the PID of the process which says ovw under the CMD column.
- 3. Look for the process whose CMD column has jre in it **and** whose PPID matches the PID of the ovw process. This is the *fvtopomap* process, highlighted in the below example.

> /usr/	/bin/ps	-elf	egrep	"PID jre	ovw"
---------	---------	------	-------	----------	------

F	S	UID	PID	PPID	С	PRI	NI	ADDR	SZ	WCHAN	STIME	TTY	TIME	CMD
8	S		1166	1160				•					•	ovhelp -ovwRunning
8	S		1164	1160				•					•	ovw navigator
8	S		29818	29817				•					•	ovwdb -0
8	S	•	235	29817	٠	•			•			•		/usr/fore/foreview/ jre/bin//bin/s
8	s	•	1162	1160	•	•	•	•	•	•	•	•	•	/usr/fore/foreview/ jre/bin//bin/s
8	S		1160	852										ovw

#### A.1.1.2 Checking Fvtopomap on Windows NT

Windows NT includes a tool called Task Manager for tracking system performance. Task Manager provides a summary of how processes such as *fvtopomap* are using CPU and memory resources. With OpenView running, use the Task Manager to find *fvtopomap* in the list of processes.

- 1. Right-click on the toolbar and then click Task Manager.
- 2. Click on the Processes tab.
- 3. Find *fvtopomap* in the list of processes, as shown in Figure A.1.

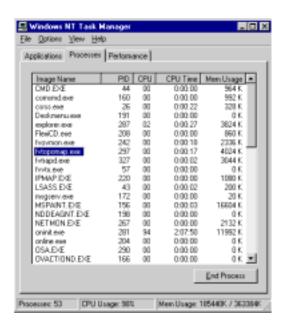


Figure A.1 - Windows NT Task Manager

### A.1.2 Checking Error Logs

All traps that are received by OpenView's *trapd* are logged in the events log, trapd.log. A subset of traps specific to FORE ATM switches are routed to *fvtrapd*. When changes in status or topology of the ATM network are not reflected in the map, check the following:

- Make sure that changes in status or topology of the network (e.g., a port became operational), are logged as traps from the switch.
- Make sure that changes of status or topology cause updating of the database. You
  should do that by looking for *fvtrapd* traps that follow traps from the switch.
- Look for changes of status (color) or topology (links to hosts or switch) on the ATM network submaps. Even if the map does not reflect the change immediately, check it after 2 minutes. That is the default interval in which *fvtopomap* checks the database to re-synchronize itself.

All errors are logged to application specific log files:

For UNIX, log files go to /usr/fore/foreview/tmp/application-name.trc.

For Windows NT, log files go to \foreview\tmp\application-name.trc

Severe errors are written to stdout in addition to being logged.

#### A.1.3 Examining the OVW Database

Another important debugging tool is *ovobjprint*, which displays the contents of the OVW database. It is useful to print the contents of the database into a file, and then to check that the ATM objects in the database represent the current ATM network topology, as well as checking that the ATM map represents the ATM database objects correctly. For more information about this tool, refer to the HP OpenView man pages and/or manuals.

### A.1.4 Deleting ForeView Map Objects

If the map does not appear the way you want it, you may want to remove the objects, symbols and submaps created by *ForeView* while leaving objects and symbols created by other applications unaffected. The *fvovdbclean* script allows you to do this. Use the following procedure to remove *ForeView* map objects:

1. Stop the *fvovmon* and *fvtrapd* applications with the following command:

#### ovstop fvovmon fvtrapd

- 2. Make sure that you have only one OpenView session running.
- 3. Switch OpenView to the window with a ForeView submap.
- 4. Issue the **fvovdbclean** from another x-term window. All ForeView symbols and submaps should disappear.
- 5. Repeat steps three and four for each OpenView map with a *ForeView* submap that you want to delete.
- 6. Restart fvovmon and fvtrapd:

#### ovstart fvovmon fvtrapd

Keep in mind the following guidelines when you use fvovdbclean:

- fvtopomap must be active
- fvovdbclean only acts on the currently active map.

To run *fvovdbclean*, execute it as a command line command in the \$FOREVIEW\_TOP/bin directory. For example:

fvovdbclean

## A.2 Use Netscape to View On-line Help

*ForeView* provides an on-line Help manual in HTML format to assist you during network management tasks. On UNIX systems, to use *ForeView* on-line help (available under the *ForeView* option of the Help menu) you must have installed Netscape on the workstation and it must be in the path. If *ForeView* can't find Netscape in the path an error is displayed.



You may experience slow response to help queries while running Netscape and ovw simultaneously.

On Windows NT, you can use either Netscape or Microsoft Internet Explorer.

## A.3 Maintaining the *ForeView* Database

### A.3.1 Removing Old Status Entries from the Database

If the ForeView database becomes full with status entries, and no longer accepts new status entry insertions, you can run the *fvtrimdb* script. This script deletes all but the most recent status entry for each node in the fv\_node\_s\_1 table and each link in the fv\_spans\_link\_s\_1, fv uni link s 1, and fv hwc link s 1 tables.

When *fvtrimdb* completes, every node or link that had at least one status entry in a table will still have one and only one entry in the table.

The *fvtrimdb* script does not affect the operation of *ForeView* in anyway except allowing additional status entry insertions.

To run *fvtrimdb*, execute it as a root command line command in the FOREVIEW\_TOP/bin directory. For example:

#### fvtrimdb

The fvtrimdb script supports the following command line switches:

- help Displays usage information.

-tracingOn Displays tracing information as the script is

executed.

- traceFile < filename > Writes tracing information to the specified file.

## A.4 Modifying Application Launch Parameters

Two mechanisms are provided to modify the launch environments of *ForeView* applications:

- On Windows NT, ForeView includes the launch.ini file that allows the user to specify unique launch requirements for ForeView applications in a single file, minimizing the install time runtime process environment configuration customization to a single registry key.
- On UNIX, *ForeView* includes a script to control/alter how the NIAMs (*fvdisco* and *fvmon*) are started when the management station is rebooted.

#### A.4.1 The Windows NT Launcher

On the Windows NT platform, *ForeView* provides both a graphical and command line method to launch *ForeView*. The graphical launcher is associated with a *ForeView* icon. The choice of using either launch method is dictated by whether a console based output of the application is desirable, or whether the launching application redirects the standard input/output/error handles.

The launcher relies on a launch definition file, launch.ini, that contains generic as well as application specific definitions required by *ForeView* applications. From the executable name, the launcher shall perform a lookup of the application launch definition file, configure the launcher process environment, then launch the application with pipe redirected standard output and standard error.

The launch definition file is encoded using the Windows NT INI profile format, and can be edited using a standard text editor. There are three sections type: Registry, Environment, and Application. There is a main section group, as well as per application section groups. The main section group contains process definitions shared by all applications. The application section group contains process definitions specific to that application.

The main section group includes [Registry.Main], and [Environment.Main]. The application sections are named [Registry.<appName>], [Environment.<appName>] and [Application.<appName>].

- The [Registry] Section section defines Windows NT registry key values required by the application.
- The [Environment] Section defines environment variables that must be set before launching the application.
- The [Application] Section defines the launch command for the application. It also contains entries for controlling the behavior of the launcher itself.

#### A.4.1.1 Example

```
[Registry.Main]
FOREVIEW_TOP=HKLM "software\FORE Systems, Inc.\ForeView" PathName
FOREVIEW_HOME=HKLM "software\FORE Systems, Inc." FOREVIEW_HOME ""
OV_MAIN_PATH=HKLM "software\Hewlett-Packard\OpenView\Network Node Manager" PathName
[Environment.Main]
FOREVIEW_TOP=$sfn($(FOREVIEW_TOP))
OV_MAIN_PATH=$sfn($(OV_MAIN_PATH))
FV_BIN=$dos($(FOREVIEW_TOP)/bin)
FV_TEMP=$dos($(FOREVIEW_TOP)/tmp)
FVJRE=$dos($(FOREVIEW_TOP)/jre/bin/jre.exe)
FVJRE_LIB=$dos($(FOREVIEW_TOP)/jre/lib)
PATH=$(FV_BIN);$nul(PATH)
FV_CLASSPATH=$dos($(FOREVIEW_TOP)/classes)
[Application.fvtrapd]
exe=$(FVJRE)
arg=-cp
arg=$(FV_CLASSPATH)
arg=-DFOREVIEW_TOP=$(FOREVIEW_TOP)
arg=com.fore.fv.ov.trapd.Fvtrapd
[Application.fvtopomap]
exe=$(FVJRE)
arg=-cp
arg=$(FV_CLASSPATH)
arg=-DFOREVIEW_TOP=$(FOREVIEW_TOP)
```

arg=-DOV\_MAIN\_PATH=\$(OV\_MAIN\_PATH)
arg=com.fore.fv.ov.topomap.Fvtopomap

### A.4.2 The UNIX Boot Script

On the UNIX platform, *ForeView* provides a script that controls how the NIAMs (*fvdisco* and *fvmon*) are started up when the management station is rebooted. The file is located in /etc/init.d/foreview, and looks like the following:

```
FOREVIEW_TOP=/usr/fore/foreview5
export FOREVIEW_TOP

case "$1" in
    start)
        test -x /usr/fore/foreview5/bin/fvstart || exit 0
        /usr/fore/foreview5/bin/fvstart disco > /dev/console 2>&1 &
        /usr/fore/foreview5/bin/fvstart mon > /dev/console 2>&1 &
        ;;
    stop)
        test -x /usr/fore/foreview5/bin/fvstop || exit 0
        /usr/fore/foreview5/bin/fvstop
        ;;
    *)
        echo "Usage: foreview [ start | stop ]"
        ;;
esac
exit 0
```

Users can edit the fystart lines to specify different startup options for fydisco and fymon.



# The *ForeView* Database Schema

This appendix introduces the use of Relational Database Management Systems (RDBMSs) with *ForeView*, and explains the structure and content of the data in the *ForeView* database (*fvdb*). The following table identifies the major components of the database schema.

**Table B.1 -** Summary of *ForeView* Data Tables

Data Type	Purpose	Refer to
Topology	Defines the tables that support the topology discovery of a <i>ForeRunner</i> ATM network, including the discovery of ATM switches, LAN Access devices, ATM-connected hosts, NNI and UNI links in an ATM network.	
Monitoring	Defines the tables that support the monitoring function. For each managed node and link in the discovery schema, the monitoring NIAM provides both current and historical status information about each entity.	
Mapping	Defines the tables that support the <i>ForeView</i> mapping process, including the mapping table that synchs the <i>ForeView</i> database with the OpenView database.	
SNMP Configuration	Defines the table that supports the configuration of SNMP agents on FORE devices.	page B-23

#### **B.1 Schema Overview**

ForeView's new architecture encompasses an underlying database that allows ForeView to scale and manage much larger networks. The database provides a central integration point for adding new capabilities, features, and partner solutions; and also serves as a central repository for discovery and monitoring records. The database schema is defined by tables, indexes, and views.

- The tables define the data and relations among the data. Each table is described
  by a collection of data types, descriptions, and an indication of the unique key for
  each table. There may be more than one column used to define a unique key.
- Indexes are separate objects that optimize the access of a table. There may be more
  than one index per table and the indexes do not always correspond with the
  unique keys.
- Views are meta-tables that are used when it is determined that frequent queries
  are joining the same tables. A view allows the query processing to take place in
  the database engine automatically, improves query performance, and simplifies
  the client side by presenting a meta table for further querying.

The discovery NIAM (*fvdisco*) determines the topology of a FORE Systems' ATM network. It discovers the ATM switches, LAN access devices, ATM-connected hosts, NNI and UNI links in the network. In the context of discovery, a network topology can be defined in terms of nodes and the links that interconnect these nodes. A node can be one of the following:

- A single ATM switch fabric
- A host
- A LAN access device such as an ES-3810.

The ports, sub-ports, adapters, etc., are the interfaces on a node. A physical or logical link connects an interface on a node to interfaces on other nodes. In other words, an interface on a node originates and terminates the links.

An interface can originate or terminate multiple links (e.g., a point-to-multipoint connection between a BUS and LECs) and a link can connect multiple (more that two) endpoints. Thus there is a many to many relationship between an interface and links. By introducing the concept of an endpoint, this relationship is simplified to two one-to-many relationships, i.e., an interface has many endpoints and a link connects two or more endpoints.

Typically a physical link connecting two nodes in the network is modeled as an entry in the  $fv_link_1$  table that connects two endpoints on the nodes. For a given interface (port) on a node, an endpoint is created for each (logical) link that connects that interface to another interface. For each link, *ForeView* provides three types of status:

- Hardware carrier status
- UNI signalling status
- SPANS signalling status

Figure B.1 illustrates the layout and relationship of the *fvdb* schema. The generic term *node* is applied to both switches, LAN access devices and hosts. A link is defined as any logical or physical connection between two or more endpoints.

The connecting lines between boxes are logical relationships between table data, and the arrowheads represent multiplicity on the table with the arrow. For example, a single entry in the fv\_node\_1 table may reference one or more entries in the fv\_address\_1 table.

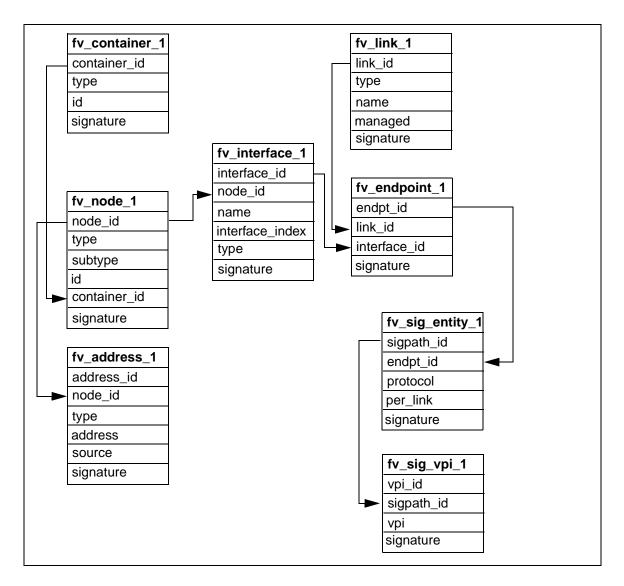
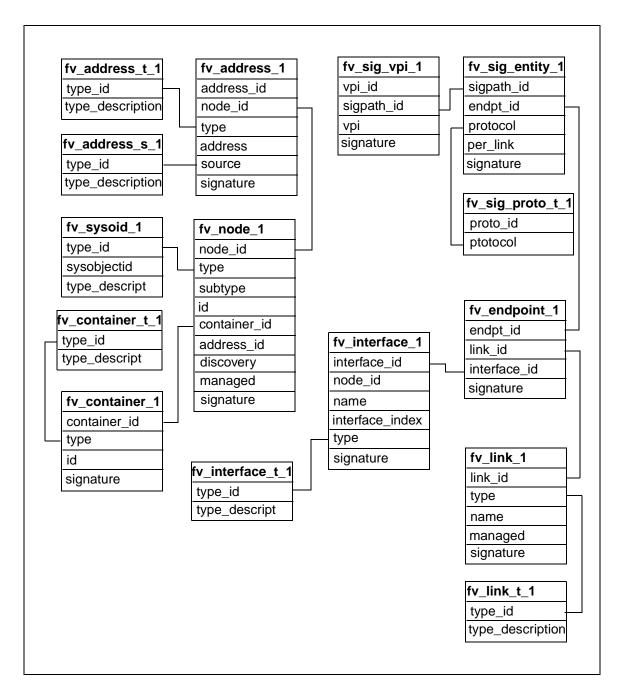


Figure B.1 - Example of the Database Relationship Schema

## **B.2 Topology Data Tables**

This section details the schema that the *ForeView* database uses to support the network discovery process. The topology of the ATM network is made up of nodes and links. A node can be a single ATM switch fabric, an ATM-connected adapter, or an ATM-connected LAN access device such as an ES-3810. A link is normally associated with two nodes (which can be two ATM switches or one ATM switch connected to an access device). The relationships that make up the topology portion of the database are illustrated in Figure B.2.



**Figure B.2 -** Topology Relations in the *ForeView* Database

#### B.2.1 The fv\_node\_1 Table

A node is a single switch fabric, a host, or a LAN access switch. A node must be uniquely identified by the combination of type, subtype and id.

An ATM Switch that encloses multiple switch fabrics, like the ASX-1000, will be represented by a row in the fv\_container table and a row for each enclosed switch fabric in the fv\_node table. Essentially an entry in the fv\_node table exists for each addressable SNMP agent in a given network device. These individual entries are grouped together by a container in the fv\_container table. The container\_id field is set to NULL for all nodes that are not enclosed in a container (e.g., ASX-200BX, ES-3810, etc.).

The combination of the type, subtype and id uniquely identify a node on the network i.e., the id field uniquely identifies a node within nodes of the same type and subtype. A unique index will be placed on the combination of type, subtype and id. This will prevent duplicate entries in the table. By querying the type, subtype and id of a given IP address, fvdisco can figure out if the IP address belongs to a node that has already been discovered. The unique id precludes the need to maintain a table of all IP addresses of the node to figure out if a given IP address belongs to a node that has already been discovered.

An entry in the fv\_node table is made only after succesfully discovering the type, subtype and id. If an unreachable IP address is discovered as the remote address of a UNI/NNI link, an entry in the fv\_node table will not be made.

The discovery field can be set to 'M' to prevent fvtopod from discovering (and/or changing) a manually added node. This field should be set to 'V' for fvdisco to discover and verify the type, subtype and id fields. Fvdisco sets it to 'A' after verification.

Table B.2 - fv\_node\_1

Column Name	Data Type	Description
node_id	INTEGER	The unique identifier for this row generated by the database server.
type	INTEGER	An integer representing the type of the node. See the fv_sysoid table.
subtype	INTEGER	An integer representing the subtype of the node. e.g., switchType for ATM switches.
id	VARCHAR(12)	A string that uniquely identifies this node. e.g., board- SerialNumber for ATM switches.
container_id	INTEGER	For any device that can enclose multiple nodes/agents this identifies the container id of the enclosure. See the fv_container table. Set to NULL for nodes that are not enclosed.
address_id	INTEGER	The preferred agent address_id to be used for SNMP agent accesses.
discovery	CHAR(1)	Values are 'A' for auto discovered entry; 'M' for manual entry (no verification); and 'V' for manual entry that should be verified by discovery.
managed	CHAR(1)	Values are 'M' for managed and 'U' for unmanaged. Only managed nodes are (re)discovered and monitored i.e., kept up to date.
signature	INTEGER	Signature of the process that last updated the row.

# **B.2.2** The fv\_container\_1 Table

Each container has a unique identifier. For an ASX-1000 this would be the envMgmtBoardSerialNumber, commonly referred to as the enclosureId. A unique index will be placed on the combination of type and id. This will prevent duplicate entries in the table.

Column Name	Data Type	Description
container_id	INTEGER	The unique identifier for this row generated by the database server.
type	INTEGER	See the fv_container_t_1 table for valid node types.
id	VARCHAR(12)	A number that uniquely identifies this container.
signature	INTEGER	Signature of the process that created the row.

**Table 5.1 -** fv\_container\_1

# B.2.3 The fv\_address\_1 Table

The fv\_address\_1 table is the single storage for all types of node or endpoint addresses. Addresses are represented as character strings. Each address in the fv\_address\_1 table will have a valid node\_id. An index is placed on type and address to prevent duplicate entries.

Column Name	Data Type	Description	
address_id	INTEGER	The unique identifier for this row generated by the database server.	
node_id	INTEGER	The node to which this address belongs. See the fv_node_1 table.	
type	INTEGER	See the fv_address_t_1 table for valid address types.	
address	VARCHAR(32)	The address.	
source	INTEGER	From where this address was discovered (e.g. a SPANS signalling path). See the fv_address_src_1 table for valid source types.	
signature	INTEGER	Signature of the process that last updated this row.	

Table B.3 - fv\_address\_1

# B.2.4 The fv\_interface\_1 Table

The fv\_interface\_1 table will have an entry for each port/sub-port of interest on a node. The fv\_interface\_1 table will have entries only for ports that either have carrier or have a link to another node. Because a node is created only after discovering its type, subtype and id., ports on which there is no carrier and no discoverable remote node are not represented in the fv\_interface\_1 table (in other words, all interfaces in the fv\_interface\_1 table either have carrier or links on them).

ATM uplinks on LAN access devices and ATM adapters in the hosts also will have entries in this table. The interface name must be unique for a node. For this reason, a unique index is placed on node\_id and name.

**Table B.4 -** fv\_interface\_1

Column Name	Data Type	Description
interface_id	INTEGER	The unique identifier for this row generated by database server.
node_id	INTEGER	The node to which this interface belongs. See the fv_node_1 table.
name	VARCHAR(32)	The logical name of this interface, typically the BNP.
interface_index	INTEGER	A value derived from the index in the MIB ifIndex table.
type	INTEGER	Type of the interface. See fv_interface_t_1 table for valid types of interfaces.
signature	INTEGER	Signature of the process that last updated this row.

# B.2.5 The fv\_endpoint\_1 Table

The fv\_endpoint\_1 table defines the endpoints for any link. A link can connect multiple endpoints so that CEM/FRAM/FUNI and IMA can be supported. An endpoint is always associated with an interface. An index is placed on link\_id and interface\_id to prevent duplicating endpoints for a link on an interface.

Column Name	Data Type	Description	
endpt_id	INTEGER	The unique identifier for this row generated by the database server.	
link_id	INTEGER	Reference to link_id from the fv_link_1 table.	
interface_id	INTEGER	Reference to the interface_id from the fv_interface_1 table.	
signature	INTEGER	Signature of the process that created this row.	

**Table B.5 -** fv\_endpoint\_1

# B.2.6 The fv\_link\_1 Table

The endpoints of a link in the  $fv_link_1$  table are defined in the  $fv_endpoint_1$  table. A link in the  $fv_link_1$  table is created only after discovering nodes on at least two ends of the link. Fvtopod will not create entries for links with undiscoverable/unreachable remote nodes.

Column Name	Data Type	Description
link_id	INTEGER	The unique identifier for this row generated by the database server.
type	INTEGER	See fv_link_t_1 for valid link types.
name	VARCHAR(64)	A unique name by which to reference the interface.
managed	CHAR(1)	Values are 'M' for managed or 'U' for unmanaged.
signature	INTEGER	Signature of the process that last updated this row.

Table B.6 - fv link 1

# B.2.7 The fv\_sig\_entity\_1 Table

This table has entries for each row of interest in the signalling path and q2931AdminTable. Notice that an entry in the fv\_sig\_entity\_1 table is associated with a single endpoint. The signalling entity may use any VPI indexed in the fv\_sig\_vpi\_1 table. If per\_link is TRUE there will be no entries in the fv\_sig\_vpi\_1 table and it may use all unused/unallocated VPIs for circuit allocations.

Column Name	Data Type	Description
sigpath_id	INTEGER	The unique identifier for this row generated by the database server.
endpt_id	INTEGER	The endpoint for this signaling entity.
protocol	INTEGER	See fv_sig_proto_t_1 for valid protocol types.
per_link	CHAR(1)	Values are 'T' for TRUE and 'F' for FALSE. The signal- ling entity may use all unallocated VPIs for circuit allo- cations when TRUE.
signature	INTEGER	Signature of the process that last updated this row.

**Table B.7 -** fv\_sig\_entity\_1

# B.2.8 The fv\_sig\_vpi\_1 Table

Typically a signalling entity uses the VPI on which it is instantiated for VCC allocation. But with the introduction of non-vp associate signalling, a signalling entity may use all of the unallocated VPI space for connections, or it may use a range of VPIs. This table is the list of VPI(s) that a signalling entity may use. Note that this table will have no entries for a signalling entity if its PER\_LINK field is set to 'T' (true).

Column Name	Data Type	Description	
vpi_id	INTEGER	Unique id for this row generated by database.	
sigpath_id	INTEGER	Reference to SIG_ENTITY_ID from the FV_SIG_ENTITY table.	
vpi	INTEGER	The VPI this signal path may operate on.	
signature	INTEGER	Signature of the process that last updated this row.	

Table B.8 - fv\_sig\_vpi\_1

# **B.2.9 Supporting Tables**

The following are List Of Value (LOV) tables that support the primary tables defined in the discovery schema. In general, these tables provide numeric identifiers with descriptive strings designed to:

- Normalize strings out of the primary tables, thereby making the tables smaller and more compact.
- Provide an easily maintained place to modify global strings for various database values.
- Provide the flexibility to build in additional string tables to allow for custom or more explanatory text per string without changing any of the values in the base schema.

### B.2.9.1 The fv container t 1 Table

This table provides a numeric index that corresponds to a specific type of enclosure for a multi-fabric switch such as an ASX-1000. The numerical value of the TYPE\_ID field is cross-referenced to the id field of the fv\_container\_1 table.

**Table B.9 -** fv container t 1

Type_ID	Description	
1	ASX-1000 Enclosure	

# B.2.9.2 The fv\_address\_t\_1 Table

This table provides a numeric index that corresponds to valid types of node or endpoint addresses. The numerical value of the TYPE\_ID field is cross-referenced to the type field of the fv\_address\_1 table.

Table B.10 - fv\_address\_t\_1

Type_ID	Description
1	IP

### B.2.9.3 The fv address src 1 Table

This table provides a numeric index that corresponds to valid discovery sources of node or endpoint addresses. The numerical value of the TYPE\_ID field is cross-referenced to the source field of the fv\_address\_1 table.

Type\_ID Description

SPANS

UNI

User Defined

Other

**Table B.11 -** fv\_address\_src\_1

### B.2.9.4 The fv interface t 1 Table

This table provides a numeric index that corresponds to a specific type of node. The numerical value of the TYPE\_ID field is cross-referenced to the type field of the fv\_interface\_1 table. The fv\_interface\_1 table will have entries only for ports that either have carrier or have a link to another node. ATM uplinks on LAN access devices and ATM adapters in the hosts also will have entries in this table.

Type\_ID Description

1 ATM Port
2 ATM Uplink

**Table B.12 -** fv interface t 1

### B.2.9.5 The fv\_link\_t\_1 Table

This table provides a numeric index that corresponds to a specific type of link. The numerical value of the TYPE\_ID field is cross-referenced to the type field of the fv\_link table.

Type_ID	Description	
1	Unknown	
2	User Defined	
3	NNI Link	
4	UNI Link	

Table B.13 - fv link t 1

### B.2.9.6 The fv\_sig\_proto\_t\_1 Table

This table provides a numeric index that corresponds to the protocol for signalling paths. The numerical value of the TYPE\_ID field is cross-referenced to the protocol field of the fv\_sig\_entity\_1 table.

 Proto\_ID
 Protocol

 1
 Unknown

 2
 SPANS

 3
 UNI

 4
 FT PNNI

 5
 ATMF PNNI

**Table B.14 -** fv\_sig\_proto\_t\_1

### B.2.9.7 The fv\_sysoid\_1 Table

This table provides a numeric index that corresponds to the object identifier for a specific device. The numerical value of the <code>TYPE\_ID</code> field is cross-referenced to the <code>type</code> field of the <code>fv\_node\_1</code> table.

Type_ID	SYSOBJECTID	Description
1	Unknown	Unknown
2	.1.3.6.1.4.1.326.2.2	FORE ATM Switch
3	.1.3.6.1.4.1.326.2.6.1.1	FORE PowerHub
4	.1.3.6.1.4.1.326.2.3	FORE ES-3810
5	.1.3.6.1.4.1.326.2.1	FORE ATM Adapter

Table B.15 -  $fv_sysoid_1$ 

# **B.2.9.8 MIB Variables for Unique IDs**

The following table defines the sources for the unique identifier for different types of nodes.

**Table B.16 - MIB** Variables for Unique ID

Type of Device	MIB Variable	Description
ASX-4000 ATM Switches	envMgmtBoardSerialNumber	Serial number of the CEC module.
ASX-1000 ATM Switches	boardSerialNumber	Serial number of the switch fabric.
ATM Switches	dot1dBaseBridgeAddress	MAC address of the Packet Engine
PowerHubs	dot1dBaseBridgeAddress	MAC address of the NNM module.
ES-3810s	envMgmtBoardSerialNumber	Serial number of the CEC module.

# **B.3 Monitoring Data Tables**

This section details the schema that the *ForeView* database uses to support the monitoring process. The monitoring schema is based on the discovery schema, which provides the monitoring NIAM with the nodes and links to monitor. For each managed node and link in the discovery schema, the monitoring NIAM provides both current and historical status information about each entity.



The monitoring NIAM will not monitor nodes with the status of unmanaged.

Only the status (up or down) for each node/link in the discovery schema is monitored in this release of *ForeView*. The status of a node/link defined in the monitoring tables is actually an extension of node/link tables in the discovery schema. The discovery and monitoring tables key on same node and link identifiers.

For status monitoring, there are four primary tables, the <u>fv\_node\_status</u> table, and three link status tables: fv\_hwc\_link\_s\_1, fv\_uni\_link\_s\_1, and fv\_spans\_link\_s\_1.

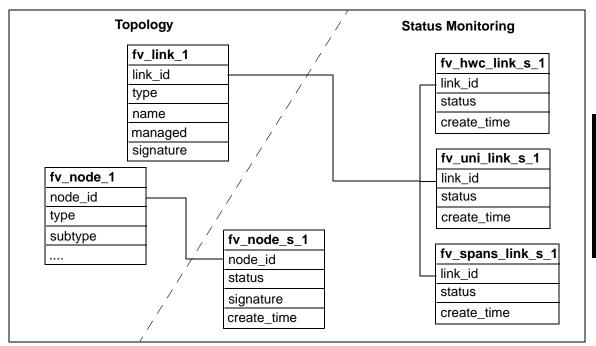


Figure B.3 - Status Monitoring Relations in the ForeView Database

# B.3.1 The fv\_node\_s\_1 Table

The fv\_node\_s\_1 table collects the status changes of any node.



The node\_id is the same attribute defined in the fv\_node\_1 table as discussed in the discovery section of this appendix.

For ATM switches, the status of a node is determined by way of accessibility. If a node is not accessible while trying to determine the link status, the node is marked down. If the node responds to any SNMP query, it is marked up.

For LAN access devices (ES-3810s, etc.), the status is determined by way of the link. The link status between a node and its neighboring switch is examined, and if there is one signalling path up, it can be inferred the device must be up.

Table B.17 - fv\_node\_s\_1

Column Name	Data Type	Description	
node_id	INTEGER	From the fv_node_1 table of the discovery schema, the unique identifier for this row generated by the database server.	
status	VARCHAR(1)	The status of the node. Valid options are 'U' for up and 'D' for down.	
signature	VARCHAR(25)	The process that created this row.	
create_time	TIME	Time stamp when the node status last changed.	

# B.3.2 The fv\_hwc\_link\_s\_1 Table

The fv\_hwc\_link\_s\_1 table collects the hardware carrier status.



The  $link\_id$  is the same attribute defined in the  $fv\_link\_1$  table as discussed in the discovery schema section of this appendix.

Table B.18 - fv\_hwc\_link\_s\_1

Column Name	Data Type	Description	
link_id	INTEGER	From the fv_link_1 table of the discovery schema, the unique identifier for this row generated by the database server.	
status	VARCHAR(1)	The status information for the MIB variable hwPortOperStatus. Valid options are 'U' for up 'D' for down, 'N' for none, and '0' for unknown.	
create_time	TIME	Time stamp when the link status last changed.	

# B.3.3 The fv\_uni\_link\_s\_1 Table

The fv\_uni\_link\_s\_1 table collects the UNI signalling status.



The  $link\_id$  is the same attribute defined in the  $fv\_link\_1$  table as discussed in the discovery schema section of this appendix.

Table B.19 - fv\_uni\_link\_s\_1

Column Name	Data Type	Description	
link_id	INTEGER	From the fv_link_1 table of the discovery schema, the unique identifier for this row generated by the database server.	
status	VARCHAR(1)	The summary information about MIB variable q2931ILMIOperStatus. If a link has more than one UNI signalling path, the summary value here will the OR of all the operational status of UNI signalling paths. Valid options are 'U' for up 'D' for down, 'N' for none, and '0' for unknown.	
create_time	TIME	Time stamp when the link status last changed.	

# B.3.4 The fv\_spans\_link\_s\_1 Table

The fv\_spans\_link\_s\_1 table collects the SPANS signalling status.



The link\_id is the same attribute defined in the fv\_link\_1 table as discussed in the discovery schema section of this appendix.

Table B.20 - fv\_spans\_link\_s\_1

Column Name	Data Type	Description
link_id	INTEGER	From the fv_link_1 table of the discovery schema, the unique identifier for this row generated by the database server.
status	VARCHAR(1)	The summary information about MIB variable sig-PathOperStatus. If a link has more than one SPANS signalling path, the summary value here will be the OR of all the operational status of SPANS signalling paths. Valid options are 'U' for up 'D' for down, 'N' for none, and '0' for unknown.
create_time	TIME	Time stamp when the link status last changed.

# **B.3.5** Supporting Tables

# B.3.5.1 The fv mon interval Table

The fv\_mon\_interval\_1 table determines how often entities within any priority actually are polled. This table is keyed on the priority\_id field.

Table B.21 - fv\_mon\_interval

Column Name	Data Type	Description	
priority_id	INTEGER	Refer to the fv_mon_priority_1 table.	
polling_interval	INTEGER	The current polling interval. The polling_interval is the average time an entity in a certain priority is polled by the monitoring NIAM. Average means this value is a moving average over last 10 minute period.	

### B.3.5.2 The fv\_mon\_priority\_1 Table

The fv\_mon\_priority\_1 table identifies any entity that should be polled at a lower or higher priority. In this case, entity means any object defined in the discovery schema such as node, link, interface, endpoint, etc.

Column Name **Description Data Type INTEGER** entity\_type\_id Refers to the entity type id value from the fy entity t 1 table. entity\_id **INTEGER** Refers to the entity type id value from the fv\_entity\_t\_1 table. A value of null means all entities with the same entity type id will have the priority defined in the PRIORITY field. **INTEGER** priority Accepts a value from 1 to 20, 1 being the highest priority, and 20 being the lowest priority.

**Table B.22 -** fv\_mon\_priority\_1

### B.3.5.3 The fv\_entity\_t\_1 Table

The fv\_entity\_t\_1 table applies a numeric index to each type of discovered entity. The fv\_mon\_priority\_1 table uses these indexes when adjusting the polling priority of a specific entity.

entity_type_id	type_description		
1	FV_NODE_SWITCH		
2	FV_NODE_LAN_ACCESS		
3	FV_LINK_NNI		
4	FV_LINK_UNI		

Table B.23 - fv\_entity\_t\_1

# **B.4 Mapping Data Tables**

This section details the schema used by the *ForeView* database to support the mapping process, which minimizes *ForeView's* interaction with the OpenView database, thus increasing response time for map updates.

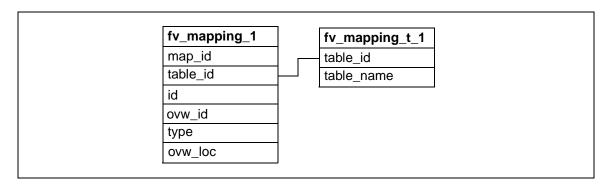


Figure B.4 - Mapping Relations in the ForeView Database

# B.4.1 The fv\_mapping\_1 Table

The fv\_mapping\_1 table maps ForeView database node\_ids and link\_ids to corresponding OpenView object identifiers.

Column	Туре	Description	
map_id	INTEGER	Unique identifier for this row generated by the database server.	
table_id	INTEGER	The index corresponding to the table_id from the fv_table_t_1 table.	
id	INTEGER	A primary key used to reference the row in the table referred to by the entry in the table_id column.	
ovw_id	INTEGER	The OpenView Object ID that corresponds to the ForeView database object defined by the table_id and id values.	

Table B.24 - fv\_mapping\_1

Table B.24 - fv\_mapping\_1

Column	Туре	Description	
type	INTEGER	The object type will be one of the following: NNISwObject, SwObject, Host, Edge, UNILink or NNILink.	
ovw_loc	INTEGER	The name of the machine where the OpenView database resides.	

# **B.4.2 Supporting Tables**

### B.4.2.1 The fv\_table\_t\_1 Table

The fv\_mapping\_t\_1 table provides unique identification indexes for the *ForeView* topology data tables. Currently there are six topology tables relevant to *ForeView*'s mapping process. As more topology tables are created this reference table can be updated.

Table B.25 - fv\_table\_t\_1

table_id	Table Name	
1	fv_node_1	
2	fv_link_1	
3	fv_container_1	
4	fv_address_1	
5	fv_interface_1	
6	fv_endpoint_1	

# **B.5 SNMP Configuration Data Table**

This section details the schema used by the *ForeView* database to support the SNMP community string actions for devices in the network.

# B.5.1 The fv\_snmp\_conf\_1 Table

The fv\_mapping\_t\_1 table records changes to the default values for agent community names, timeout and retry intervals, proxies, remote ports, and polling intervals.

Table B.26 - fv\_snmp\_conf\_1

Column	Туре	Description	
row_id	INTEGER	Unique identifier for this row generated by the database server.	
name	VARCHAR(64)	The device name used to reference the device by application programs.	
community	VARCHAR(128)	The read community name.	
setcommunity	VARCHAR(128)	The write community name.	
proxy	VARCHAR(64)	The host that will proxy the SNMP request, either the host name or IP address.	
timeout	INTEGER	The amount of time, in seconds, for a response to an SNMP query. If a response is not received within the specified time interval, the request is resent.	
retry	INTEGER	The number of times a query to a target device will be resent.	
pollinterval	INTEGER	Not applicable for this release.	
remoteport	INTEGER	The UPD port on the target device.	

# **B.6 Querying the Database**

The major components of the *ForeView* database are the tables for Topology, Monitoring, Mapping, and SNMP Configuration. This section provides example queries that might be of interest to a user. You can perform queries using any SQL tools supported by the RDBMS in use.

### CAUTION



The *ForeView* database tables are considered read-only. Writing to the tables is not supported and could corrupt the data.

# **B.6.1** Query for Discovered Nodes

The following is an example of a query of all the nodes discovered.

# **B.6.2** Query for the Number of Monitored Nodes

The following is an example of a query of the number of monitored nodes.

# **B.6.3** Query for the Number of Links

The following is an example of a query of the number of links.

# B.6.4 Query for the Number of NNI/UNI/HWC Links

The following is an example of a query of the number of NNI/UNI/HWC links.

# B.6.5 Query to List the Links by Port and IP Address

The following is an example of an advanced query to output a list of the links by port and IP address.

```
SELECT A1.ADDRESS||' '||I1.NAME||'<-->'||I2.NAME||' '||A2.ADDRESS AS LINK
FROM
        fv_link_1 L,
        fv_endpoint_1 E1, fv_interface_1 I1, fv_node_1 N1, fv_address_1 A1,
                fv_sysoid_1 S1,
        fv_endpoint_1 E2, fv_interface_1 I2, fv_node_1 N2, fv_address_1 A2,
                fv_sysoid_1 S2
WHERE
        E1.LINK_ID = L.LINK_ID AND
        L.TYPE != 5 AND
        i1.INTERFACE_ID = E1.INTERFACE_ID AND
        A1.NODE_ID = I1.NODE_ID AND
        N1.NODE_ID = I1.NODE_ID AND
        A1.TYPE = 1 AND
        N1.TYPE = S1.TYPE_ID AND
        E2.LINK_ID = L.LINK_ID AND
        12.INTERFACE_ID = E2.INTERFACE_ID AND
        A2.NODE_ID = I2.NODE_ID AND
        N2.NODE_ID = I2.NODE_ID AND
        A2.TYPE = 1 AND
        N2.TYPE = S2.TYPE_ID AND
        n1.node_id <= n2.node_id AND i1.interface_id != i2.interface_id AND
        A1.ADDRESS_ID = N1.ADDRESS_ID AND
        A2.ADDRESS_ID = N2.ADDRESS_ID
        ORDER BY 1
```

```
Sample output would be:

link 172.24.67.34 1C1<-->1A2 172.24.64.33

link 172.24.67.34 1C2<-->ATM_UPLINK 172.24.66.41

link 172.24.67.34 1C3<-->4A1 172.24.64.36

3 row(s) retrieved.
```

# **B.6.6** Query for the Status of the Links

The following is an example of an advanced query to list the status of the links.

```
SELECT 1.link_id AS LINK, h.status AS HARDWARE,
       u.status AS UNI, s.status AS SPANS
FROM
        fv_link_1 l, OUTER fv_hwc_link_s_1 h, OUTER fv_uni_link_s_1 u,
        OUTER fv_spans_link_s_1 s
WHERE
        h.link_id = l.link_id AND
        h.create_time =
                ( SELECT MAX( create_time )
                  FROM fv_hwc_link_s_1
                  WHERE link_id = l.link_id ) AND
        u.link_id = l.link_id AND
        u.create_time =
                ( SELECT MAX( create_time )
                  FROM fv_uni_link_s_1
                  WHERE link_id = l.link_id ) AND
        s.link_id = l.link_id AND
        s.create time =
                ( SELECT MAX( create_time )
                  FROM fv_spans_link_s_1
                  WHERE link_id = l.link_id )
Sample output:
       link hardware uni spans
          1 U
                    U N
          2 U
          3 U
3 row(s) retrieved.
```

# **B.6.7 Table Dumps**

You can dump the contents of database tables by using the DBaccess utility. To invoke it, type dbaccess. Choose a command by moving the cursor using a left or right arrow key and then hitting return, or by typing the capital letter in the command name (usually the first letter).

### B.6.7.1 fv\_node\_1

The following is an example of the output from the fv\_node\_1 table.

```
node_id
type
            1
subtype
           10
           32770
container_id
address_id 1
discovery A
managed
           M
signature
node_id 2
type
           2
subtype
id
           00A036008D34
container_id
address_id 3
discovery A
managed
           M
signature
node_id 3
type
subtype
          5
           15056
container_id
address_id 4
discovery
          Α
managed
          M
signature
23 row(s) retrieved.
```

# B.6.7.2 fv\_link\_1

The following is an example of the output from the fv\_link\_1 table.

link_id	type	name	managed	signature
1	4	1	M	
2	3	2	M	
3	3	3	M	

<sup>3</sup> row(s) retrieved.

# B.6.7.3 fv\_endpoint\_1

The following is an example of the output from the fv\_endpoint\_1 table.

endpt_id	link_id	interface_id	signature
1	1	2	
2	1	4	
3	2	1	
4	2	5	
5	3	3	
6	3	14	

<sup>6</sup> row(s) retrieved.

# B.6.7.4 fv\_interface\_1

The following is an example of the output from the  $fv_interface_1$  table.

de 1C1 derface_index 16 de 1 mature  derface_id 2 de_id 1 de 1C2 derface_index 17 de 1 mature  derface_index 3	interface_id	1				
erface_index 16 be 1 mature  erface_id 2 le_id 1 lee 1C2 erface_index 17 be 1 mature  erface_id 3	node_id	1				
mature  erface_id 2 le_id 1 lee 1C2 lerface_index 17 lee 1 lemature  erface_id 3	name	1C1				
erface_id 2 de_id 1 de 1C2 derface_index 17 de 1 mature  derface_id 3	interface_index	16				
erface_id 2 le_id 1 le 1C2 erface_index 17 le 1 mature  derface_id 3	type	1				
de_id 1 de 1C2 derface_index 17 de 1 mature derface_id 3	signature					
de_id 1 de 1C2 derface_index 17 de 1 mature derface_id 3						
de 1C2 derface_index 17 de 1 mature derface_id 3	interface_id	2				
erface_index 17 be 1 mature  erface_id 3	node_id	1				
erface_id 3	name	1C2				
mature erface_id 3	interface_index	17				
erface_id 3	type	1				
_	signature					
_						
	interface_id	3				
ie_ia I	node_id	1				

### The ForeView Database Schema

```
1C3
name
interface_index 18
                1
type
signature
interface_id 4
node_id
name
              ATM_UPLINK
interface_index 1
type
                2
signature
interface_id 5
node_id
                3
name
              1A2
interface_index 1
               1
type
signature
interface_id 6
node_id
               3
name
              1A4
interface_index 3
type
                1
signature
interface_id 51
node_id
              17
name
               ATM_UPLINK
interface_index 1
type
signature
interface_id
               52
node_id
              18
name
               ATM_UPLINK
interface_index 1
type
signature
26 row(s) retrieved.
```

### B.6.7.5 fv\_node\_s\_1

The following is an example of the output from the fv\_node\_s\_1 table.

node_id	status	signature	create_time		
_					
1	U		1998-09-23	08:30:32.000	
4	U		1998-09-23	08:35:52.000	
3	U		1998-09-23	08:35:53.000	
2	U		1998-09-23	08:35:58.000	
10	U		1998-09-23	08:38:45.000	
11	U		1998-09-23	08:38:45.000	
16	U		1998-09-23	09:01:49.000	
17	U		1998-09-23	09:01:55.000	
18	U		1998-09-23	09:01:59.000	
19	U		1998-09-23	09:02:02.000	

11 row(s) retrieved.

### B.6.7.6 fv hwc link s 1

The following is an example of the output from the fv\_hwc\_link\_s\_1 table.

```
link_id status create_time
```

```
2 U 1998-09-23 08:35:52.000
3 U 1998-09-23 08:35:56.000
1 U 1998-09-23 08:35:58.000
```

3 row(s) retrieved.

# B.6.7.7 fv\_uni\_link\_s\_1

The following is an example of the output from the fv\_uni\_link\_s\_1 table.

```
link_id status create_time
```

```
1 U 1998-09-23 08:31:46.000
3 U 1998-09-23 08:35:56.000
2 U 1998-09-23 08:31:47.000
```

3 row(s) retrieved.

# B.6.7.8 fv\_spans\_link\_s\_1

The following is an example of the output from the  $fv\_spans\_link\_s\_1$  table.

link\_id status create\_time

```
2 U 1998-09-23 08:35:52.000
3 U 1998-09-23 08:35:56.000
1 N 1998-09-23 08:35:57.000
```

3 row(s) retrieved.

# icensing Agreement

# **APPENDIX C** Licensing Agreements

This appendix lists the legal statements for various shareware software items that are used in the *ForeView* 5.0 release. Their use requires the inclusions of these statements. *ForeView* 5.0 uses the following software:

- JRE (Java Runtime Environment)
- Apache (Web Server)
- JGL (ObjectSpace Java Class Library)
- Perl (Scripting language for install)

# C.1 Java(tm) Runtime Environment

### Version 1.1.5

### **Binary Code License**

This binary code license ("License") contains rights and restrictions associated with use of the accompanying Java Runtime Environment software and documentation ("Software"). Read the License carefully before using the Software. By using the Software you agree to the terms and conditions of this License.

- 1. License to Distribute. Licensee is granted a royalty-free right to reproduce and distribute the Software provided that Licensee: (i) distributes the Software complete and unmodified (except for the specific files identified as optional in the Software README file), only as part of, and for the sole purpose of running, Licensee's Java compatible applet or application ("Program") into which the Software is incorporated; (ii) does not distribute additional software intended to replace any component(s) of the Software; (iii) agrees to incorporate the most current version of the Software that was available 180 days prior to each production release of the Program; (iv) does not remove or alter any proprietary legends or notices contained in the Software; (v) includes the provisions of Sections 2, 3, 5, 6, 8 and 9 in Licensee's license agreement for the Program; and (vi) agrees to indemnify, hold harmless, and defend Sun and its licensors from and against any claims or lawsuits, including attorneys' fees, that arise or result from the use or distribution of the Program.
- 2. Java Platform Interface. Licensee may not modify the Java Platform Interface ("JPI", identified as classes contained within the "java" package or any subpackages of the "java" package), by creating additional classes within the JPI or otherwise causing the addition to or modification of the classes in the JPI. In the event that Licensee creates any Java-related API and distributes such API to others for application development, Licensee must promptly publish broadly, an accurate specification for such API for free use by all developers of Java-based software.
- 3. Restrictions. Software is confidential copyrighted information of Sun and title to all copies is retained by Sun and/or its licensors. Licensee shall not decompile, disassemble, decrypt, extract, or otherwise reverse engineer Software. Software may not be leased, assigned, or sublicensed, in whole or in part, except as specifically authorized in Section 1. Software is not designed or intended for use in online control of aircraft, air traffic, aircraft navigation or aircraft communications; or in the design, construction, operation or maintenance of any nuclear facility. Licensee warrants that it will not use or redistribute the Software for such purposes.
- 4. Trademarks and Logos. This License does not authorize Licensee to use any Sun name, trademark or logo. Licensee acknowledges that Sun owns the Java trade-

mark and all Java-related trademarks, logos and icons including the Coffee Cup and Duke ("Java Marks") and agrees to: (i) comply with the Java Trademark Guidelines at >>http://java.sun.com/trademarks.html; (ii) not do anything harmful to or inconsistent with Sun's rights in the Java Marks; and (iii) assist Sun in protecting those rights, including assigning to Sun any rights acquired by Licensee in any Java Mark.

- Disclaimer of Warranty. Software is provided "AS IS," without a warranty of any kind. ALL EXPRESS OR IMPLIED REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE HEREBY EXCLUDED.
- 6. Limitation of Liability. SUN AND ITS LICENSORS SHALL NOT BE LIABLE FOR ANY DAMAGES SUFFERED BY LICENSEE OR ANY THIRD PARTY AS A RESULT OF USING OR DISTRIBUTING SOFTWARE. IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF THE USE OF OR INABILITY TO USE SOFTWARE, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
- 7. Termination. This license shall automatically terminate 180 days after production release of the next version of the Software by Sun. Licensee may terminate this License at any time by destroying all copies of Software. This License will terminate immediately without notice from Sun if Licensee fails to comply with any provision of this License. Upon such termination, Licensee must destroy all copies of Software.
- 8. Export Regulations. Software, including technical data, is subject to U.S. export control laws, including the U.S. Export Administration Act and its associated regulations, and may be subject to export or import regulations in other countries. Licensee agrees to comply strictly with all such regulations and acknowledges that it has the responsibility to obtain licenses to export, re-export, or import Software. Software may not be downloaded, or otherwise exported or re-exported (i) into, or to a national or resident of, Cuba, Iraq, Iran, North Korea, Libya, Sudan, Syria or any country to which the U.S. has embargoed goods; or (ii) to anyone on the U.S. Treasury Department's list of Specially Designated Nations or the U.S. Commerce Department's Table of Denial Orders.
- 9. Restricted Rights. Use, duplication or disclosure by the United States government is subject to the restrictions as set forth in the Rights in Technical Data and Computer Software Clauses in DFARS 252.227-7013(c) (1) (ii) and FAR 52.227-19(c) (2) as applicable.
- 10. Governing Law. Any action related to this License will be governed by California

- law and controlling U.S. federal law. No choice of law rules of any jurisdiction will apply.
- 11. Severability. If any of the above provisions are held to be in violation of applicable law, void, or unenforceable in any jurisdiction, then such provisions are herewith waived or amended to the extent necessary for the License to be otherwise enforceable in such jurisdiction. However, if in Sun's opinion deletion or amendment of any provisions of the License by operation of this paragraph unreasonably compromises the rights or increase the liabilities of Sun or its licensors, Sun reserves the right to terminate the License.

# C.2 Apache Group

Copyright (c) 1995-1998 The Apache Group. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- 3. All advertising materials mentioning features or use of this software must display the following acknowledgment: "This product includes software developed by the Apache Group for use in the Apache HTTP server project (http://www.apache.org/)."
- 4. The names "Apache Server" and "Apache Group" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.
- 5. Products derived from this software may not be called "Apache" nor may "Apache" appear in their names without prior written permission of the Apache Group.
- 6. Redistributions of any form whatsoever must retain the following acknowledgment: "This product includes software developed by the Apache Group for use in the Apache HTTP server project (http://www.apache.org/)."

THIS SOFTWARE IS PROVIDED BY THE APACHE GROUP "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE GROUP OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Group and was originally based on public domain software written at the National Center for Supercomputing Applications, University of Illinois, Urbana-Champaign.

For more information on the Apache Group and the Apache HTTP server project, please see <a href="http://www.apache.org/">http://www.apache.org/</a>.

# C.3 JGL - The Generic Collection Library for Java(tm)

Release Version 2.0

WHEREAS, ObjectSpace, Inc. is the owner of valuable intellectual property rights relating to the Generic Library Release for Java(tm) Version 2.0 ("JGL") and wishes to license JGL subject to the terms and conditions set forth below; and WHEREAS, you ("Licensee") acknowledge that ObjectSpace, Inc. has the right to grant licenses to the intellectual property rights relating to JGL, and that you desire to obtain a license to use JGL subject to the terms and conditions set forth below;

ObjectSpace, Inc. grants Licensee a non-exclusive, non-transferable, royalty-free license to use JGL and related materials without charge provided the Licensee adheres to all of the terms and conditions of this Agreement.

By downloading, using, or copying JGL or any portion thereof, Licensee agrees to abide by the intellectual property laws and all other applicable laws of the United States of America, and to all of the terms and conditions of this Agreement, and agrees to take all necessary steps to ensure that the terms and conditions of this Agreement are not violated by any person or entity under the Licensee's control or in the Licensee's service.

Licensee shall maintain the copyright and trademark notices on the materials within or otherwise related to JGL, and not alter, erase, deface or overprint any such notice.

Except as specifically provided in this Agreement, Licensee is expressly prohibited from copying, modifying, merging, selling, leasing, assigning, or transferring in any manner, JGL or any portion thereof.

Licensee may copy materials within or otherwise related to JGL that bear the ObjectSpace copyright only as required for backup purposes or for use solely by the Licensee.

Licensee may not distribute in any form of electronic or printed communication the materials within or otherwise related to JGL that bear the ObjectSpace copyright, including but not limited to the source code, documentation, help files, examples, and benchmarks, without prior written consent from ObjectSpace, Inc. Send any requests for limited distribution rights to jgl@objectspace.com.

The Licensee may distribute binaries (.class files) derived from or contained within JGL ("Binaries") provided that:

1) The Binaries are not integrated, bundled, combined, or otherwise associated with a Java development environment or Java development tool; and 2) The Binaries are not a documented part of any distribution material.

Licensing

OBJECTSPACE, INC. DISCLAIMS AND MAKES NO REPRESENTATIONS OR WARRANTIES ABOUT THE SUITABILITY OF JGL, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. OBJECTSPACE, INC. SHALL NOT BE LIABLE FOR ANY DIRECT OR CONSEQUENTIAL DAMAGES SUFFERED BY LICENSEE AS A RESULT OF USING, MODIFYING, OR DISTRIBUTING JGL OR ITS DERIVATIVES.

# C.4 PERL Script

This agreement shall be construed and enforced in accordance with the laws of the United States of America and the State of Texas (excluding its choice of law rules).

The ``Artistic License"

### **Preamble**

The intent of this document is to state the conditions under which a Package may be copied, such that the Copyright Holder maintains some semblance of artistic control over the development of the package, while giving the users of the package the right to use and distribute the Package in a more-or-less customary fashion, plus the right to make reasonable modifications.

### **Definitions**

- "Package" refers to the collection of files distributed by the Copyright Holder, and derivatives of that collection of files created through textual modification.
- ``Standard Version'' refers to such a Package if it has not been modified, or has been modified in accordance with the wishes of the Copyright Holder as specified below.
- ``Copyright Holder" is whoever is named in the copyright or copyrights for the package.
- ``You" is you, if you're thinking about copying or distributing this Package.
- `Reasonable copying fee" is whatever you can justify on the basis of media cost, duplication charges, time of people involved, and so on. (You will not be required to justify it to the Copyright Holder, but only to the computing community at large as a market that must bear the fee.)
- `Freely Available" means that no fee is charged for the item itself, though there may be fees involved in handling the item. It also means that recipients of the item may redistribute it under the same conditions they received it.
  - You may make and give away verbatim copies of the source form of the Standard Version of this Package without restriction, provided that you duplicate all of the original copyright notices and associated disclaimers.
  - You may apply bug fixes, portability fixes and other modifications derived from the Public Domain or from the Copyright Holder. A Package modified in such a way shall still be considered the Standard Version.
  - 3. You may otherwise modify your copy of this Package in any way, provided that you insert a prominent notice in each changed file stating how and when you changed that file, and provided that you do at least ONE of the following:
    - a. place your modifications in the Public Domain or otherwise make them Freely Available, such as by posting said modifications to Usenet or an equivalent medium, or placing the modifications on a major archive site

- such as uunet.uu.net, or by allowing the Copyright Holder to include your modifications in the Standard Version of the Package.
- b. use the modified Package only within your corporation or organization.
- c. rename any non-standard executables so the names do not conflict with standard executables, which must also be provided, and provide a separate manual page for each non-standard executable that clearly documents how it differs from the Standard Version.
- d. make other distribution arrangements with the Copyright Holder.
- 4. You may distribute the programs of this Package in object code or executable form, provided that you do at least ONE of the following:
  - a. distribute a Standard Version of the executables and library files, together with instructions (in the manual page or equivalent) on where to get the Standard Version.
  - b. accompany the distribution with the machine-readable source of the Package with your modifications.
  - c. give non-standard executables non-standard names, and clearly document the differences in manual pages (or equivalent), together with instructions on where to get the Standard Version.
  - d. make other distribution arrangements with the Copyright Holder.
- 5. You may charge a reasonable copying fee for any distribution of this Package. You may charge any fee you choose for support of this Package. You may not charge a fee for this Package itself. However, you may distribute this Package in aggregate with other (possibly commercial) programs as part of a larger (possibly commercial) software distribution provided that you do not advertise this Package as a product of your own. You may embed this Package's interpreter—within an executable of yours (by linking); this shall be construed as a mere form of aggregation, provided that the complete Standard Version of the interpreter is so embedded.
- 6. The scripts and library files supplied as input to or produced as output from the programs of this Package do not automatically fall under the copyright of this Package, but belong to whomever generated them, and may be sold commercially, and may be aggregated with this Package. If such scripts or library files are aggregated with this Package via the so-called "undump" or "unexec" methods of producing a binary executable image, then distribution of such an image shall neither be construed as a distribution of this Package nor shall it fall under the restrictions of Paragraphs 3 and 4, provided that you do not represent such an executable image as a Standard Version of this Package.
- 7. C subroutines (or comparably compiled subroutines in other languages) supplied by you and linked into this Package in order to emulate subroutines and variables of the language defined by this Package shall not be considered part of this Package, but are the equivalent of input as in Paragraph 6, provided these subroutines

### Licensing Agreements

do not change the language in any way that would cause it to fail the regression tests for the language.

- 8. Aggregation of this Package with a commercial distribution is always permitted provided that the use of this Package is embedded; that is, when no overt attempt is made to make this Package's interfaces visible to the end user of the commercial distribution. Such use shall not be construed as a distribution of this Package.
- 9. The name of the Copyright Holder may not be used to endorse or promote products derived from this software without specific prior written permission.
- 10. THIS PACKAGE IS PROVIDED ``AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTIBILITY AND FITNESS FOR A PARTICULAR PURPOSE.

# C.5 ODBC Driver Manager License

Microsoft ODBC 3.0 Driver Manager

END-USER LICENSE AGREEMENT FOR MICROSOFT SOFTWARE

IMPORTANT—READ CAREFULLY: This Microsoft End-User License Agreement ("EULA") is a legal agreement between you (either an individual or a single entity) and Microsoft Corporation for the Microsoft software product identified above, which includes computer software and may include associated media, printed materials, and "online" or electronic documentation ("SOFTWARE PRODUCT"). By installing, copying, or otherwise using the SOFTWARE PRODUCT, you agree to be bound by the terms of this EULA. If you do not agree to the terms of this

EULA, do not install, copy or use the SOFTWARE PRODUCT.

## Software PRODUCT LICENSE

The SOFTWARE PRODUCT is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties. The SOFTWARE PRODUCT is licensed, not sold.

1. GRANT OF LICENSE. This EULA grants you the following rights

Installation and Use. You may install and use an unlimited number of copies of the SOFTWARE PRODUCT.

Reproduction and Distribution. You may reproduce and distribute an unlimited number of copies of the SOFTWARE PRODUCT; provided that each copy shall be a true and complete copy, including all copyright and trademark notices, and shall be accompanied by a copy of this EULA. Copies of the SOFTWARE

PRODUCT may be distributed as a standalone product (however, you may not charge for this) or included with your own product (you may charge for this).

2. DESCRIPTION OF OTHER RIGHTS AND LIMITATIONS.

Limitations on Reverse Engineering, Decompilation, and Disassembly. You may not reverse engineer, decompile, or disassemble the SOFTWARE PRODUCT, except and only to the extent that such activity is expressly permitted by applicable law notwithstanding this limitation.

Separation of Components. The SOFTWARE PRODUCT is licensed as a single product. Its component parts may not be separated for use on more than one computer.

Support Services. Microsoft may provide you with support services related to the SOFTWARE PRODUCT ("Support Services"). Use of Support Services is governed by the Microsoft policies and programs described in the user manual, in "online" documentation, and/or in other Microsoft-provided materials. Any supplemental

software code provided to you as part of the Support Services shall be considered part of the SOFTWARE PRODUCT and subject to the terms and conditions of this EULA. With respect to technical information you provide to Microsoft as part of the Support Services, Microsoft may use such information for its business purposes, including for product support and development. Microsoft will not utilize such technical information in a form that personally identifies you.

Software Transfer. You may permanently transfer all of your rights under this EULA, provided the recipient agrees to the terms of this EULA.

Termination. Without prejudice to any other rights, Microsoft may terminate this EULA if you fail to comply with the terms and conditions of this EULA. In such event, you must destroy all copies of the SOFTWARE PRODUCT and all of its component parts.

- 3. COPYRIGHT. All title and copyrights in and to the SOFTWARE PRODUCT (including but not limited to any images, photographs, animations, video, audio, music, text, and "applets" incorporated into the SOFTWARE PRODUCT), the accompanying printed materials, and any copies of the SOFTWARE PRODUCT are owned by Microsoft or its suppliers. The SOFTWARE PRODUCT is protected by copyright laws and international treaty provisions. Therefore, you must treat the SOFTWARE PRODUCT like any other copyrighted material.
- 4. U.S. GOVERNMENT RESTRICTED RIGHTS. The SOFTWARE PRODUCT and documentation are provided with RESTRICTED RIGHTS. Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 or subparagraphs (c)(1) and (2) of the Commercial Computer Software—Restricted Rights at 48 CFR 52.227-19, as applicable. Manufacturer is Microsoft Corporation/One Microsoft Way/Redmond, WA 98052-6399.

#### **MISCELLANEOUS**

If you acquired this product in the United States, this EULA is governed by the laws of the State of Washington.

If you acquired this product in Canada, this EULA is governed by the laws of the Province of Ontario, Canada. Each of the parties hereto irrevocably attorns to the jurisdiction of the courts of the Province of Ontario and further agrees to commence any litigation which may arise hereunder in the courts located in the Judicial District of York, Province of Ontario.

If this product was acquired outside the United States, then local law may apply.

Should you have any questions concerning this EULA, or if you desire to contact Microsoft for any reason, please contact the Microsoft subsidiary serving your country, or write: Microsoft Sales Information Center/One Microsoft Way/Redmond, WA 98052-6399.

# LIMITED WARRANTY

NO WARRANTIES. Microsoft expressly disclaims any warranty for the SOFTWARE PROD-UCT. The SOFTWARE PRODUCT and any related documentation is provided "as is" without warranty of any kind, either express or implied, including, without limitation, the implied warranties or merchantability, fitness for a particular purpose, or noninfringement. The entire risk arising out of use or performance of the SOFTWARE PRODUCT remains with you.

NO LIABILITY FOR DAMAGES. In no event shall Microsoft or its suppliers be liable for any damages whatsoever (including, without limitation, damages for loss of business profits, business interruption, loss of business information, or any other pecuniary loss) arising out of the use of or inability to use this Microsoft product, even if Microsoft has been advised of the possibility of such damages. Because some states/jurisdictions do not allow the exclusion or limitation of liability for consequential or incidental damages, the above limitation may not apply to you.

# Licensing Agreements

# **Acronyms**

The networking terms in the following list are defined in the Glossary of this manual. Glossary items are listed alphabetically according to the full term.

AAL ATM Adaptation Layer
ABR Available Bit Rate

**ACM** Address Complete Message

ACR Allowable Cell Rate

**ADPCM** Adaptive Differential Pulse Code Modulation

AHFG ATM-attached Host Functional Group

AIMUX ATM Inverse Multiplexing
AIS Alarm Indication Signal
AMI Alternate Mark Inversion
AMI ATM Management Interface

ANSI American National Standards Institute
APCM Adaptive Pulse Code Modulation
API Application Program Interface

APP Application Program

APS Automatic Protection Switching
ARP Address Resolution Protocol

**ASCII** American Standard Code for Information Interchange

**ATDM** Asynchronous Time Division Multiplexing

ATM Asynchronous Transfer Mode
AUI Attachment User Interface
BBZS Bipolar 8 Zero Substitution

**BCOB** Broadband Connection Oriented Bearer

BCOB-A Bearer Class A
BCOB-C Bearer Class C
BCOB-X Bearer Class X

**BECN** Backward Explicit Congestion Notification

**BER** Bit Error Rate

BES Bursty Errored SecondsBGP Border Gateway ProtocolB-ISDN Inter-Carrier Interface.

BIP Bit Interleaved Parity

**B-ISDN** Broadband Integrated Services Digital Network

**B-ISUP** Broadband ISDN User's Part

#### Acronyms

BITS Building Integrated Timing Supply

BPDU Bayonet-Neill-Concelman
Bridge Protocol Data Unit

bps Bits per SecondBPV Bipolar Violation

B-TE Broadband Terminal Equipment
BUS Broadcast and Unknown Server
CAC Connection Admission Control
CAS Channel Associated Signaling

CBDS Connectionless Broadband Data Service

**CBR** Constant Bit Rate

**CCITT** International Telephone and Telegraph Consultative Committee

**CCS** Common Channel Signaling

CDV Cell Delay Variation
CE Connection Endpoint

CEI Connection Endpoint Identifier
CES Circuit Emulation Service
CGA Carrier Group Alarm

CIP Carrier Identification Parameter
CIR Committed Information Rate

CLIP Classical IP
CLP Cell Loss Priority
CLR Cell Loss Ratio-1-15
CLS Connectionless service

**CMIP** Common Management Interface Protocol

**CMR** Cell Misinsertion Rate

**CPE** Customer Premise Equipment

CRA Cell Rate Adaptation
CRC Cyclic Redundancy Check

CRS Cell Relay Service
CS Controlled Slip, or

CSU Channel Service Unit
CTD Cell Transfer Delay

CTS Clear To Send

Digital Access and C

DACS Digital Access and Cross-Connect System
DARPA Defense Advanced Research Projects Agency

DCC Data Country Code

DCE Data Communications Equipment
DCS Digital Cross-connect System
DES Destination End Station

**DFA** DXI Frame Address

**DLCI** Data Link Connection Identifier

**DNS** Domain Naming System

**DSn** Digital Standard n (n=0, 1, 1C, 2, and 3)

**DSR** Data Set Ready

DTE Data Terminal Equipment
DTR Data Terminal Ready

**EEPROM** Electrically Erasable Programmable Read Only Memory

**EFCI** Explicit Forward Congestion Indication

**EGP** Exterior Gateway Protocol

**EIA** Electronics Industries Association

**EISA** Extended Industry Standard Architecture

EMI Emulated Local Area Network Electromagnetic Interference

**EPROM** Erasable Programmable Read Only Memory

EQL Equalization

ER Explicit Rate

ES End System, or

Errored Second

**ESF** Extended Super Frame **ESI** End System Identifier

**EXZ** Excessive Zeroes (Error Event)

**FC** Face Contact

**FCC** Federal Communications Commission

FCS Frame Check Sequence

FDDI Fiber Distributed Data Interface
FDM Frequency Division Multiplexing

FEE Far End Block Error
FEC Forward Error Correction

**FECN** Forward Explicit Congestion Notification

FERF Far End Receive Failure
FIFO First-In, First-Out
FRS Frame-Relay Service
FTP File Transfer Protocol
FT-PNNI ForeThought PNNI
FUNI Frame-Based UNI

GCAC Generic Connection Admission Control

GCRA Generic Cell Rate Algorithm

GFC Generic Flow Control HDB3 High Density Bipolar

**HDLC** High Level Data Link Control

**HEC** Header Error Control

**HIPPI** High Performance Parallel Interface

**HSSI** High-Speed Serial Interface

ICMP Internet Control Message Protocol

## Acronyms

**IDU** Interface Data Unit

**IEEE** Institute of Electrical and Electronics Engineers

IETF Internet Engineering Task Force
ILMI Interim Local Management Interface

IP Internet Protocol

IPX Internetwork Packet Exchange

IS Intermediate system

ISDN Integrated Services Digital Network
ISO International Standards Organization

ITU-T International Telecommunication Union Telecommunication

IWF Interworking Function IXC Interexchange Carriers

JPEG Joint Photographic Experts Group

Kbps Kilobits per second
LAN Local Area Network
LAN Emulation

LAPB Link Access Procedure, Balanced LATA Local Access and Transport Area

LEO Line Build Out
LCV Line Code Violations

LE\_ARP LAN Emulation Address Resolution Protocol

**LEC** LAN Emulation Client

**LECS** LAN Emulation Configuration Server

LES LAN Emulation Server
LLC Logical Link Control
LOF Loss Of Frame
LOP Loss Of Pointer

LOS Loss Of Signal
LSB Least Significant Bit
MAC Media Access Control
MAN Metropolitan Area Network
MAU Media Attachment Unit
MBS Maximum Burst Size

MCDV Maximum Cell Delay Variance
MCLR Maximum Cell Loss Ratio

MCR Minimum Cell Rate

MCTDMaximum Cell Transfer DelayMIBManagement Information BaseMICMedia Interface Connector

MID Message Identifier

MMF Multimode Fiber Optic Cable
MPEG Motion Picture Experts Group
MPOA Multiprotocol over ATM

MSB Most Significant Bit

MTU Maximum Transmission Unit
NM Network Management Entity
NML Network Management Layer
NMS Network Management Station

NNI Network-to-Network Interface or Network Node Interface

NPC Network Parameter Control

NRZ Non Return to Zero

NRZI Non Return to Zero Inverted
NSAP Network Service Access Point
NTSC National TV Standards Committee
OAM Operation and Maintenance Cell

OC-n Optical Carrier level-n
OID Object Identifier
OOF Out-of-Frame

OSI Open Systems Interconnection
OSPF Open Shortest Path First Protocol
OUI Organizationally Unique Identifier
PAD Packet Assembler Disassembler

PAL Phase Alternate Line
PBX Private Branch Exchange

**PCI** Peripheral Component Interconnect

**PCM** Pulse Code Modulation

PCR Peak Cell Rate

PDN Public Data Network
PDU Protocol Data Unit
PHY Physical Layer

ping Packet Internet Groper

**PLCP** Physical Layer Convergence Protocol

PLP Packet Level Protocol
PM Physical Medium

PMD Physical Medium Dependent

PNNI Private Network Node Interface or Private Network-to-Network Interface

**PPP** Point-to-Point Protocol

**PROM** Programmable Read-Only Memory

PRS Primary Reference Source
PSN Packet Switched Network

PT Payload Type

PVC Permanent Virtual Circuit (or Channel)
PVCC Permanent Virtual Channel Connection
PVPC Permanent Virtual Path Connection

QD Queuing Delay
QoS Quality of Service

## Acronyms

RD Routing Domain
RFCs Requests For Comment
RFI Radio Frequency Interference
RIP Routing Information Protocol
RISC Reduced Instruction Set Computer

RTS Request To Send
SA Source Address
SA Source MAC Address
SAP Service Access Point

SAR Segmentation And Reassembly

SC Structured Cabling, or

Structured Connectors, or

Stick and Click

**SCR** Sustainable Cell Rate

SCSI Small Computer Systems Interface
SDLC Synchronous Data Link Control

**SDU** Service Data Unit

SEAL Simple and Efficient Adaptation Layer
SECAM Systeme En Coleur Avec Memoire

SEL Selector

**SES** Severely Errored Seconds

**SF** Super Frame

**SGMP** Simple Gateway Management Protocol

**SIR** Sustained Information Rate

**SLIP** Serial Line IP

SMDS Switched Multimegabit Data Service

**SMF** Single Mode Fiber

SMTP Simple Mail Transfer Protocol
SNA Systems Network Architecture
SNAP SubNetwork Access Protocol
SNI Subscriber Network Interface

**SNMP** Simple Network Management Protocol

**SONET** Synchronous Optical Network

**SPANS** Simple Protocol for ATM Network Signalling

**SPARC** Scalable Processor Architecture Reduced instruction set Computer

**SPE** Synchronous Payload Envelope

SPVC Smart PVC

SS7 Signaling System No. 7

**SSCOP** Service Specific Connection Oriented Protocol

SSCS Service Specific Convergence Sublayer

ST Straight Tip, or

Stick and Turn

**STM** Synchronous Transfer Mode

STP Shielded Twisted Pair, Spanning Tree Protocol

STS Synchronous Transport Signal

SVC Switched Virtual Circuit (or Channel)
SVCC Switched Virtual Channel Connection
SVPC Switched Virtual Path Connection

TAXI Transparent Asynchronous Transmitter/Receiver Interface

TC Transmission Convergence
TCP Transmission Control Protocol

TCP/IP Transmission Control Protocol/Internet Protocol

TCR Tagged Cell Rate

TCS Transmission Convergence Sublayer

**TDM** Time Division Multiplexing

TE Terminal Equipment

**TFTP** Trivial File Transfer Protocol

TM Traffic Management
UAS Unavailable Seconds
UBR Unspecified Bit Rate
UDP User Datagram Protocol
UNI User-to-Network Interface
UPC Usage Parameter Control

**UTOPIA** Universal Test & Operations Interface for ATM

UTP Unshielded Twisted Pair

VBR Variable Bit Rate

VC Virtual Channel (or Circuit)
VCC Virtual Channel Connection
VCI Virtual Channel Identifier
VCL Virtual Channel Link
VINES Virtual Network Software
VLAN Virtual Local Area Network

**VP** Virtual Path

VPC Virtual Path Connection
VPDN Virtual Private Data Network

VPI Virtual Path Identifier
VPL Virtual Path Link
VPN Virtual Private Network
VPT Virtual Path Terminator

**VS/VD** Virtual Source/Virtual Destination

VT Virtual Tributary WAN Wide-Area Network

**ZBTSI** Zero Byte Time Slot Interchange

Acronyms

# **Glossary**

**10Base-T -** a 10 Mbps baseband Ethernet specification utilizing twisted-pair cabling (Category 3, 4, or 5). 10BaseT, which is part of the IEEE 802.3 specification, has a distance limit of approximately 100 meters per segment.

**802.1d Spanning Tree Bridging -** the IEEE standard for bridging; a MAC layer standard for transparently connecting two or more LANs (often called subnetworks) that are running the same protocols and cabling. This arrangement creates an extended network, in which any two workstations on the linked LANs can share data.

**802.3 Ethernet -** the IEEE standard for Ethernet; a physical-layer standard that uses the CSMA/CD access method on a bus-topology LAN.

**802.5 Token Ring -** the IEEE physical-layer standard that uses the token-passing access method on a ring-topology LAN.

**AAL Connection -** an association established by the AAL between two or more next higher layer entities.

**Adapter -** A fitting that supplies a passage between two sets of equipment when they cannot be directly interconnected.

**Adaptive Differential Pulse Code Modulation (ADPCM)** - A technique that allows analog voice signals to be carried on a 32K bps digital channel. Sampling is done at 8Hz with 4 bits used to describe the difference between adjacent samples.

**Adaptive Pulse Code Modulation (APCM) -** A technique that effectively reduces occupied bandwidth per active speaker by reducing sampling rates during periods of overflow peak traffic.

**Address -** A unique identity of each network station on a LAN or WAN.

 $\label{lem:Address Complete Message (ACM) - A B-ISUP call control message from the receiving exchange to sending exchange indicating the completion of address information.}$ 

**Address Mask** - a bit mask used to identify which bits in an address (usually an IP address) are network significant, subnet significant, and host significant portions of the complete address. This mask is also known as the subnet mask because the subnetwork portion of the address can be determined by comparing the binary version of the mask to an IP address in that subnet. The mask holds the same number of bits as the protocol address it references.

**Address Prefix -** A string of 0 or more bits up to a maximum of 152 bits that is the lead portion of one or more ATM addresses.

**Address Resolution -** The procedure by which a client associates a LAN destination with the ATM address of another client or the BUS.

**Address Resolution Protocol (ARP) -** a method used to resolve higher level protocol addressing (such as IP) into the appropriate header data required for ATM; i.e., port, VPI, and VCI; also defines the AAL type to be used.

**Agent -** a component of network- and desktop-management software, such as SNMP, that gathers information from MIBs.

**alarm -** an unsolicited message from a device, typically indicating a problem with the system that requires attention.

Alarm Indication Signal (AIS) - In T1, an all ones condition used to alert a receiver that its incoming signal (or frame) has been lost. The loss of signal or frame is detected at the receiving end, and the failed signal is replaced by all the ones condition which the receiver interprets as an AIS. The normal response to this is AIS is for the receiving end to generate a yellow alarm signal as part of its transmission towards the faulty end. (The AIS itself is sometimes called a Blue Signal).

**A-Law -** The PCM coding and companding standard used in Europe.

Allowable Cell Rate (ACR) - parameter defined by the ATM Forum for ATM traffic management. ACR varies between the MCR and the PCR, and is dynamically controlled using congestion control mechanisms.

**Alternate Mark Inversion (AMI) -** A line coding format used on T1 facilities that transmits ones by alternate positive and negative pulses.

**Alternate Routing -** A mechanism that supports the use of a new path after an attempt to set up a connection along a previously selected path fails.

American National Standards Institute (ANSI) - a private organization that coordinates the setting and approval of some U.S. standards. It also represents the United States to the International Standards Organization.

American Standard Code for Information Interchange (ASCII) - a standard character set that (typically) assigns a 7-bit sequence to each letter, number, and selected control characters.

**AppleTalk** - a networking protocol developed by Apple Computer for communication between Apple's products and other computers. Independent of the network layer, AppleTalk runs on LocalTalk, EtherTalk and TokenTalk.

**Application Layer -** Layer seven of the ISO reference model; provides the end-user interface.

**Application Program (APP) -** a complete, self-contained program that performs a specific function directly for the user.

**Application Program Interface (API) -** a language format that defines how a program can be made to interact with another program, service, or other software; it allows users to develop custom interfaces with FORE products.

**Assigned Cell -** a cell that provides a service to an upper layer entity or ATM Layer Management entity (ATMM-entity).

**asxmon -** a FORE program that repeatedly displays the state of the switch and its active ports.

**Asynchronous Time Division Multiplexing (ATDM) -** a multiplexing technique in which a transmission capability is organized into a priori, unassigned time slots. The time slots are assigned to cells upon request of each application's instantaneous real need.

**Asynchronous Transfer Mode (ATM) -** a transfer mode in which the information is organized into cells. It is asynchronous in the sense that the recurrence of cells containing information from an individual user is not necessarily periodic.

**ATM Adaptation Layer (AAL) -** the AAL divides user information into segments suitable for packaging into a series of ATM cells. AAL layer types are used as follows:

**AAL-1** - constant bit rate, time-dependent traffic such as voice and video

AAL-2 - still undefined; a placeholder for variable bit rate video transmission

**AAL-3/4 -** variable bit rate, delay-tolerant data traffic requiring some sequencing and/or error detection support (originally two AAL types, connection-oriented and connectionless, which have been combined)

**AAL-5 -** variable bit rate, delay-tolerant, connection-oriented data traffic requiring minimal sequencing or error detection support

ATM Address - Defined in the UNI Specification as 3 formats, each having 20 bytes in length.

**ATM Forum -** an international non-profit organization formed with the objective of accelerating the use of ATM products and services through a rapid convergence of interoperability specifications. In addition, the Forum promotes industry cooperation and awareness.

**ATM Inverse Multiplexing (AIMUX) -** A device that allows multiple T1 or E1 communications facilities to be combined into a single broadband facility for the transmission of ATM cells.

**ATM Layer link -** a section of an ATM Layer connection between two adjacent active ATM Layer entities (ATM-entities).

**ATM Link -** a virtual path link (VPL) or a virtual channel link (VCL).

**ATM Management Interface (AMI) -** the user interface to FORE Systems' *ForeThought* switch control software (SCS). AMI lets users monitor and change various operating configurations of FORE Systems switches and network module hardware and software, IP connectivity, and SNMP network management.

**ATM Peer-to-Peer Connection -** a virtual channel connection (VCC) or a virtual path connection (VPC) directly established, such as workstation-to-workstation. This setup is not commonly used in networks.

**ATM Traffic Descriptor -** a generic list of parameters that can be used to capture the intrinsic traffic characteristics of a requested ATM connection.

**ATM User-to-User Connection -** an association established by the ATM Layer to support communication between two or more ATM service users (i.e., between two or more next higher layer entities or between two or more ATM entities). The communication over an ATM Layer connection may be either bidirectional or unidirectional. The same Virtual Channel Identifier (VCI) is used for both directions of a connection at an interface.

**atmarp -** a FORE program that shows and manipulates ATM ARP entries maintained by the given device driver. This is also used to establish PVC connections.

**ATM-attached Host Functional Group (AHFG) -** The group of functions performed by an ATM-attached host that is participating in the MPOA service.

atmconfig - a FORE program used to enable or disable SPANS signaling.

**atmstat -** a FORE program that shows statistics gathered about a given adapter card by the device driver. These statistics include ATM layer and ATM adaptation layer cell and error counts. This can also be used to query other hosts via SNMP.

Attachment User Interface (AUI) - IEEE 802.3 interface between a media attachment unit (MAU) and a network interface card (NIC). The term AUI can also refer to the rear panel port to which an AUI cable might attach.

**Auto-logout -** a feature that automatically logs out a user if there has been no user interface activity for a specified length of time.

**Automatic Protection Switching (APS) -** Equipment installed in communications systems to detect circuit failures and automatically switch to redundant, standby equipment.

**Available Bit Rate (ABR) -** a type of traffic for which the ATM network attempts to meet that traffic's bandwidth requirements. It does not guarantee a specific amount of bandwidth and the end station must retransmit any information that did not reach the far end.

**Backbone** - the main connectivity device of a distributed system. All systems that have connectivity to the backbone connect to each other, but systems can set up private arrangements with each other to bypass the backbone to improve cost, performance, or security.

**Backplane -** High-speed communications line to which individual components are connected.

**Backward Explicit Congestion Notification (BECN) -** A Resource Management cell type generated by the network or the destination, indicating congestion or approaching congestion for traffic flowing in the direction opposite that of the BECN cell.

**Bandwidth -** usually identifies the capacity or amount of data that can be sent through a given circuit; may be user-specified in a PVC.

**Baud** - unit of signalling speed, equal to the number of discrete conditions or signal events per second. If each signal event represents only one bit, the baud rate is the same as bps; if each signal event represents more than one bit (such as a dibit), the baud rate is smaller than bps.

**Bayonet-Neill-Concelman (BNC) -** a bayonet-locking connector used to terminate coaxial cables. BNC is also referred to as Bayonet Network Connector.

**Bipolar 8 Zero Substitution (B8ZS)** - a technique used to satisfy the ones density requirements of digital T-carrier facilities in the public network while allowing 64 Kbps clear channel data. Strings of eight consecutive zeroes are replaced by an eight-bit code representing two intentional bipolar pulse code violations (000V10V1).

**Bipolar Violation (BPV) -** an error event on a line in which the normal pattern of alternating high (one) and low (zero) signals is disrupted. A bipolar violation is noted when two high signals occur without an intervening low signal, or vice versa.

**B-ISDN Inter-Carrier Interface (B-ICI) -** An ATM Forum defined specification for the interface between public ATM networks to support user services across multiple public carriers.

Bit Error Rate (BER) - A measure of transmission quality, generally shown as a negative exponent, (e.g.,  $10^{-7}$  which means 1 out of  $10^7$  bits [1 out of 10,000,000 bits] are in error).

Bit Interleaved Parity (BIP) - an error-detection technique in which character bit patterns are forced into parity, so that the total number of one bits is always odd or always even. This is accomplished by the addition of a one or zero bit to each byte, as the byte is transmitted; at the other end of the transmission, the receiving device verifies the parity (odd or even) and the accuracy of the transmission.

Bit Robbing - The use of the least significant bit per channel in every sixth frame for signaling.

**Bit Stuffing -** A process in bit-oriented protocols where a zero is inserted into a string of ones by the sender to prevent the receiver from interpreting valid user data (the string of ones) as control characters (a Flag character for instance).

**Border Gateway Protocol (BGP) -** used by gateways in an internet connecting autonomous networks. It is derived from experiences learned using the EGP.

**bps** - bits per second

**Bridge -** a device that expands a Local Area Network by forwarding frames between data link layers associated with two separate cables, usually carrying a common protocol. Bridges can usually be made to filter certain packets (to forward only certain traffic).

**Bridge Protocol Data Unit (BPDU) -** A message type used by bridges to exchange management and control information.

**Broadband -** a service or system requiring transmission channels capable of supporting rates greater than the Integrated Services Digital Network (ISDN) primary rate.

**Broadband Access -** an ISDN access capable of supporting one or more broadband services.

**Broadband Connection Oriented Bearer (BCOB) -** Information in the SETUP message that indicates the type of service requested by the calling user.

**BCOB-A (Bearer Class A) -** Indicated by ATM end user in SETUP message for connection-oriented, constant bit rate service. The network may perform internetworking based on AAL information element (IE).

**BCOB-C (Bearer Class C) -** Indicated by ATM end user in SETUP message for connection-oriented, variable bit rate service. The network may perform internetworking based on AAL information element (IE).

**BCOB-X (Bearer Class X) -** Indicated by ATM end user in SETUP message for ATM transport service where AAL, traffic type and timing requirements are transparent to the network.

**Broadband Integrated Services Digital Network (B-ISDN) -** a common digital network suitable for voice, video, and high-speed data services running at rates beginning at 155 Mbps.

**Broadband ISDN User's Part (B-ISUP) -** A protocol used to establish, maintain and release broadband switched network connections across an SS7/ATM network.

**Broadband Terminal Equipment (B-TE) -** An equipment category for B-ISDN which includes terminal adapters and terminals.

**Broadcast** - Data transmission to all addresses or functions.

Broadcast and Unknown Server (BUS) - in an emulated LAN, the BUS is responsible for accepting broadcast, multicast, and unknown unicast packets from the LECs to the broadcast MAC address (FFFFFFFFFF) via dedicated point-to-point connections, and forwarding the packets to all of the members of the ELAN using a single point-to-multipoint connection.

**Brouter (bridging/router) -** a device that routes some protocols and bridges others based on configuration information.

**Buffer -** A data storage medium used to compensate of a difference in rate of data flow or time of occurrence of events when transmitting data from one device to another.

**Building Integrated Timing Supply (BITS)** - a master timing supply for an entire building, which is a master clock and its ancillary equipment. The BITS supplies DS1 and/or composite clock timing references for synchronization to all other clocks and timing sources in that building.

**Bursty Errored Seconds (BES) -** a BES contains more than 1 and fewer than 320 path coding violation error events, and no severely errored frame or AIS defects. Controlled slips are not included in determining BESs.

**Bursty Second -** a second during which there were at least the set number of BES threshold event errors but fewer than the set number of SES threshold event errors.

Byte - A computer-readable group of bits (normally 8 bits in length).

**Call -** an association between two or more users or between a user and a network entity that is established by the use of network capabilities. This association may have zero or more connections.

**Carrier -** a company, such as any of the "baby Bell" companies, that provide network communications services, either within a local area or between local areas.

Carrier Group Alarm (CGA) - A service alarm generated by a channel bank when an out-of-frame (OOF) condition exists for some predetermined length of time (generally 300 milliseconds to 2.5 seconds). The alarm causes the calls using a trunk to be dropped and trunk conditioning to be applied.

**Carrier Identification Parameter (CIP) -** A 3 or 4 digit code in the initial address message identifying the carrier to be used for the connection.

**cchan -** a FORE program that manages virtual channels on a *ForeRunner* switch running asxd.

**Cell -** an ATM Layer protocol data unit (PDU). The basic unit of information transported in ATM technology, each 53-byte cell contains a 5-byte header and a 48-byte payload.

**Cell Delay Variation (CDV) -** a quantification of cell clumping for a connection. The cell clumping CDV (yk) is defined as the difference between a cell's expected reference arrival time (ck) and its actual arrival time (ak). The expected reference arrival time (ck) of cell k of a specific connection is max. T is the reciprocal of the negotiated peak cell rate.

**Cell Delineation** - the protocol for recognizing the beginning and end of ATM cells within the raw serial bit stream.

**Cell Header -** ATM Layer protocol control information.

**Cell Loss Priority (CLP)** - the last bit of byte four in an ATM cell header; indicates the eligibility of the cell for discard by the network under congested conditions. If the bit is set to 1, the cell may be discarded by the network depending on traffic conditions.

**Cell Loss Ratio -** In a network, cell loss ratio is (1-x/y), where y is the number of cells that arrive in an interval at an ingress of the network; and x is the number of these y cells that leave at the egress of the network element.

**Cell Loss Ratio (CLR) -** CLR is a negotiated QoS parameter and acceptable values are network specific. The objective is to minimize CLR provided the end-system adapts the traffic to the changing ATM layer transfer characteristics. The Cell Loss Ratio is defined for a connection as: Lost Cells/Total Transmitted Cells. The CLR parameter is the value of CLR that the network agrees to offer as an objective over the lifetime of the connection. It is expressed as an order of magnitude, having a range of 10-1 to 10-15 and unspecified.

**Cell Misinsertion Rate (CMR)** - the ratio of cells received at an endpoint that were not originally transmitted by the source end in relation to the total number of cells properly transmitted.

**Cell Rate Adaptation (CRA)** - a function performed by a protocol module in which empty cells (known as unassigned cells) are added to the output stream. This is because there always must be a fixed number of cells in the output direction; when there are not enough cells to transmit, unassigned cells are added to the output data stream.

**Cell Relay Service (CRS)** - a carrier service which supports the receipt and transmission of ATM cells between end users in compliance with ATM standards and implementation specifications.

**Cell Transfer Delay -** the transit delay of an ATM cell successfully passed between two designated boundaries. See CTD.

**Cell Transfer Delay (CTD) -** This is defined as the elapsed time between a cell exit event at the measurement point 1 (e.g., at the source UNI) and the corresponding cell entry event at the measurement point 2 (e.g., the destination UNI) for a particular connection. The cell transfer delay between two measurement points is the sum of the total inter-ATM node transmission delay and the total ATM node processing delay.

**Channel -** A path or circuit along which information flows.

**Channel Associated Signaling (CAS) -** a form of circuit state signaling in which the circuit state is indicated by one or more bits of signaling status sent repetitively and associated with that specific circuit.

**Channel Bank -** A device that multiplexes many slow speed voice or data conversations onto high speed link and controls the flow.

**Channel Service Unit (CSU) -** An interface for digital leased lines which performs loopback testing and line conditioning.

**Channelization -** capability of transmitting independent signals together over a cable while still maintaining their separate identity for later separation.

**Circuit -** A communications link between points.

**Circuit Emulation Service (CES) -** The ATM Forum circuit emulation service interoperability specification specifies interoperability agreements for supporting Constant Bit Rate (CBR) traffic over ATM networks that comply with the other ATM Forum interoperability agreements. Specifically, this specification supports emulation of existing TDM circuits over ATM networks.

Classical IP (CLIP) - IP over ATM which conforms to RFC 1577.

**Clear to Send (CTS) -** and RS-232 modem interface control signal (sent from the modem to the DTE on pin 5) which indicates that the attached DTE may begin transmitting; issuance in response to the DTE's RTS.

**Clocking - Regularly timed impulses.** 

**Closed User Group -** A subgroup of network users that can be its own entity; any member of the subgroup can only communicate with other members of that subgroup.

**Coaxial Cable -** Coax is a type of electrical communications medium used in the LAN environment. This cable consists of an outer conductor concentric to an inner conductor, separated from each other by insulating material, and covered by some protective outer material. This medium offers large bandwidth, supporting high data rates with high immunity to electrical interference and a low incidence of errors. Coax is subject to distance limitations and is relatively expensive and difficult to install.

**Cold Start Trap -** an SNMP trap which is sent after a power-cycle (see *trap*).

**Collision -** Overlapping transmissions that occur when two or more nodes on a LAN attempt to transmit at or about the same time.

**Committed Information Rate (CIR) -** CIR is the information transfer rate which a network offering Frame Relay Services (FRS) is committed to transfer under normal conditions. The rate is averaged over a minimum increment of time.

**Common Channel Signaling (CCS) -** A form signaling in which a group of circuits share a signaling channel. Refer to SS7.

**Common Management Interface Protocol (CMIP) -** An ITU-TSS standard for the message formats and procedures used to exchange management information in order to operate, administer maintain and provision a network.

**Concatenation -** The connection of transmission channels similar to a chain.

**Concentrator -** a communications device that offers the ability to concentrate many lower-speed channels into and out of one or more high-speed channels.

**Configuration -** The phase in which the LE Client discovers the LE Service.

**Congestion Management -** traffic management feature that helps ensure reasonable service for VBR connections in an ATM network, based on a priority, sustained cell rate (SCR), and peak cell rate (PCR). During times of congestion, bandwidth is reduced to the SCR, based on the priority of the connection.

**Connection -** the concatenation of ATM Layer links in order to provide an end-to-end information transfer capability to access points.

**Connection Admission Control (CAC) -** the procedure used to decide if a request for an ATM connection can be accepted based on the attributes of both the requested connection and the existing connections.

Connection Endpoint (CE) - a terminator at one end of a layer connection within a SAP.

**Connection Endpoint Identifier (CEI) -** an identifier of a CE that can be used to identify the connection at a SAP.

**Connectionless Broadband Data Service (CBDS) -** A connectionless service similar to Bellcore's SMDS defined by European Telecommunications Standards Institute (ETSI).

**Connectionless Service -** a type of service in which no pre-determined path or link has been established for transfer of information, supported by AAL 4.

**Connectionless Service (CLS) -** A service which allows the transfer of information among service subscribers without the need for end-to- end establishment procedures.

**Connection-Oriented Service -** a type of service in which information always traverses the same pre-established path or link between two points, supported by AAL 3.

**Constant Bit Rate (CBR) -** a type of traffic that requires a continuous, specific amount of bandwidth over the ATM network (e.g., digital information such as video and digitized voice).

**Controlled Slip (CS)** - a situation in which one frame's worth of data is either lost or replicated. A controlled slip typically occurs when the sending device and receiving device are not using the same clock.

**Convergence Sublayer (CS)** - a portion of the AAL. Data is passed first to the CS where it is divided into rational, fixed-length packets or PDUs (Protocol Data Units). For example, AAL 4 processes user data into blocks that are a maximum of 64 kbytes long.

**Corresponding Entities -** peer entities with a lower layer connection among them.

cpath - a FORE program used to manage virtual paths on a ForeRunner switch running asxd.

**cport -** a FORE program that monitors and changes the state of ports on a *ForeRunner* switch running asxd.

**Cross Connection -** a mapping between two channels or paths at a network device.

**Customer Premise Equipment (CPE) -** equipment that is on the customer side of the point of demarcation, as opposed to equipment that is on a carrier side. See also point of demarcation.

**Cut Through** - Establishment of a complete path for signaling and/or audio communications.

**Cyclic Redundancy Check (CRC)** - an error detection scheme in which a number is derived from the data that will be transmitted. By recalculating the CRC at the remote end and comparing it to the value originally transmitted, the receiving node can detect errors.

D3/D4 - Refers to compliance with AT&T TR (Technical Reference) 62411 definitions for coding, supervision, and alarm support. D3/D4 compatibility ensures support of digital PBXes, M24 services, Megacom services, and Mode 3 D3/D4 channel banks at DS-1 level.

**D4 Channelization -** refers to compliance with AT&T Technical Reference 62411 regarding DS1 frame layout (the sequential assignment of channels and time slot numbers within the DS1).

**D4** Framed/Framing Format - in T1, a 193-bit frame format in which the 193rd bit is used for framing and signaling information (the frame/framing bit). To be considered in support of D4 Framing, a device must be able to synchronize and frame-up on the 193rd bit.

**Data Communications Equipment (DCE) -** a definition in the RS232C standard that describes the functions of the signals and the physical characteristics of an interface for a communication device such as a modem.

**Data Country Code (DCC) -** This specifies the country in which an address is registered. The codes are given in ISO 3166. The length of this field is two octets. The digits of the data country code are encoded in Binary Coded Decimal (BCD) syntax. The codes will be left justified and padded on the right with the hexadecimal value "F" to fill the two octets.

**Data Link** - Communications connection used to transmit data from a source to a destination.

**Data Link Connection Identifier (DLCI) -** connection identifier associated with frame relay packets that serves the same functions as, and translates directly to, the VPI/VCI on an ATM cell.

**Data Link Layer -** Layer 2 of the OSI model, responsible for encoding data and passing it to the physical medium. The IEEE divides this layer into the LLC (Logical Link Control) and MAC (Media Access Control) sublayers.

**Data Set Ready (DSR) -** an RS-232 modem interface control signal (sent from the modem to the DTE on pin 6) which indicates that the modem is connected to the telephone circuit. Usually a prerequisite to the DTE issuing RTS.

**Data Terminal Equipment (DTE) -** generally user devices, such as terminals and computers, that connect to data circuit-terminating equipment. They either generate or receive the data carried by the network.

**Data Terminal Ready (DTR) -** an RS232 modem interface control signal (sent from the DTE to the modem on pin 20) which indicates that the DTE is ready for data transmission and which requests that the modem be connected to the telephone circuit.

**Datagram -** a packet of information used in a connectionless network service that is routed to its destination using an address included in the datagram's header.

**DECnet -** Digital Equipment Corporation's proprietary LAN.

**Defense Advanced Research Projects Agency (DARPA) -** the US government agency that funded the ARPANET.

**Demultiplexing -** a function performed by a layer entity that identifies and separates SDUs from a single connection to more than one connection (see *multiplexing*).

**Destination End Station (DES) -** An ATM termination point which is the destination for ATM messages of a connection and is used as a reference point for ABR services. See SES.

**Digital Access and Cross-Connect System (DACS) -** Digital switching system for routing T1 lines, and DS-0 portions of lines, among multiple T1 ports.

**Digital Cross-connect System (DCS)** - an electronic patch panel used to route digital signals in a central office.

**Digital Standard n (0, 1, 1C, 2, and 3) (DSn) -** a method defining the rate and format of digital hierarchy, with asynchronous data rates defined as follows:

DS0	64kb/s	1 voice channel	
DS1	1.544Mb/s	24 DS0s	
DS1C	3.152 Mb/s	2 DS1s	
DS2	6.312 Mb/s	4 DS1s	
DS3	44.736 Mb/s	28 DS1s	

Synchronous data rates (SONET) are defined as:

STS-1/OC-1	51.84 Mb/s	28 DS1s or 1 DS3
STS-3/OC-3	155.52 Mb/s	3 STS-1s byte interleaved
STS-3c/OC-3c	155.52 Mb/s	Concatenated, indivisible payload
STS-12/OC-12	622.08 Mb/s	12 STS-1s, 4 STS-3cs, or any mixture
STS-12c/OC-12c	622.08 Mb/s	Concatenated, indivisible payload
STS-48/OC-48	2488.32 Mb/s	48 STS-1s, 16 STS-3cs, or any mixture

**DIP (Dual In-line Package) Switch -** a device that has two parallel rows of contacts that let the user switch electrical current through a pair of those contacts to on or off. They are used to reconfigure components and peripherals.

**Domain Name Server -** a computer that converts names to their corresponding Internet numbers. It allows users to telnet or FTP to the name instead of the number.

**Domain Naming System (DNS) -** the distributed name and address mechanism used in the Internet.

**Duplex -** Two way communication.

**DXI -** a generic phrase used in the full names of several protocols, all commonly used to allow a pair of DCE and DTE devices to share the implementation of a particular WAN protocol. The protocols define the packet formats used to transport data between DCE and DTE devices.

**DXI Frame Address (DFA) -** a connection identifier associated with ATM DXI packets that serves the same functions as, and translates directly to, the VPI/VCI on an ATM cell.

**Dynamic Allocation -** A technique in which the resources assigned for program execution are determined by criteria applied at the moment of need.

**E.164** - A public network addressing standard utilizing up to a maximum of 15 digits. ATM uses E.164 addressing for public network addressing.

**E1 -** Wide-area digital transmission scheme used predominantly in Europe that carries data at a rate of 2.048 Mbps. E1 lines can be leased for private use from common carriers.

**E3 -** Wide-area digital transmission scheme used predominantly in Europe that carries data at a rate of 34.368 Mbps. E3 lines can be leased for private use from common carriers.

**Edge Device -** A physical device which is capable of forwarding packets between legacy interworking interfaces (e.g., Ethernet, Token Ring, etc.) and ATM interfaces based on data-link and network layer information but which does not participate in the running of any network layer routing protocol. An Edge Device obtains forwarding descriptions using the route distribution protocol.

**elarp -** a FORE program that shows and manipulates MAC and ATM address mappings for LAN Emulation Clients (LECs).

**elconfig** - a FORE program that shows and modifies LEC configuration. Lets the user set the NSAP address of the LAN Emulation Configuration Server, display the list of Emulated LANs configured in the LECS for this host, display the list of ELANs locally configured along with the membership state of each, and locally administer ELAN membership.

**Electrically Erasable Programmable Read Only Memory (EEPROM) -** an EPROM that can be cleared with electrical signals rather than the traditional ultraviolet light.

**Electromagnetic Interference (EMI) -** signals generated and radiated by an electronic device that cause interference with radio communications, among other effects.

**Electronics Industries Association (EIA) -** a USA trade organization that issues its own standards and contributes to ANSI; developed RS-232. Membership includes USA manufacturers.

**Embedded SNMP Agent -** an SNMP agent can come in two forms: embedded or proxy. An embedded SNMP agent is integrated into the physical hardware and software of the unit.

**Emulated Local Area Network (ELAN) -** A logical network initiated by using the mechanisms defined by LAN Emulation. This could include ATM and legacy attached end stations.

**End System (ES)** - a system where an ATM connection is terminated or initiated (an originating end system initiates the connection).

**End System Identifier (ESI) -** This identifier distinguishes multiple nodes at the same level in case the lower level peer group is partitioned.

**End-to-End Connection -** when used in reference to an ATM network, a connection that travels through an ATM network, passing through various ATM devices and with endpoints at the termination of the ATM network.

**Enterprise -** Terminology generally referring to customers with multiple, non-contiguous geographic locations.

**Equalization (EQL)** - the process of compensating for line distortions.

**Erasable Programmable Read Only Memory (EPROM) -** A PROM which may be erased and rewritten to perform new or different functions (normally done with a PROM burner).

**Errored Second (ES)** - a second during which at least one code violation occurred.

**Ethernet -** a 10-Mbps, coaxial standard for LANs in which all nodes connect to the cable where they contend for access.

**Excessive Zeroes (EXZ) Error Event -** An Excessive Zeroes error event for an AMI-coded signal is the occurrence of more than fifteen contiguous zeroes. For a B8ZS coded signal, the defect occurs when more than seven contiguous zeroes are detected.

**Explicit Forward Congestion Indication (EFCI)** - the second bit of the payload type field in the header of an ATM cell, the EFCI bit indicates network congestion to receiving hosts. On a congested switch, the EFCI bit is set to "1" by the transmitting network module when a certain number of cells have accumulated in the network module's shared memory buffer. When a cell is received that has its EFCI bit set to "1," the receiving host notifies the sending host, which should then reduce its transmission rate.

**Explicit Rate (ER) -** The Explicit Rate is an RM-cell field used to limit the source ACR to a specific value. It is initially set by the source to a requested rate (such as PCR). It may be subsequently reduced by any network element in the path to a value that the element can sustain. ER is formatted as a rate.

**Extended Industry Standard Architecture (EISA) -** bus architecture for desktop computers that provides a 32-bit data passage and maintains compatibility with the ISA or AT architecture.

**Extended Super Frame (ESF)** - a T1 framing format that utilizes the 193rd bit as a framing bit, but whose Superframe is made up of 24 frames instead of 12 as in D4 format. ESF also provides CRC error detection and maintenance data link functions.

**Exterior Gateway Protocol (EGP) -** used by gateways in an internet, connecting autonomous networks.

**Fairness -** related to Generic Flow Control, fairness is defined as meeting all of the agreed quality of service requirements by controlling the order of service for all active connections.

**Far End Block Error (FEBE) -** an error detected by extracting the 4-bit FEBE field from the path status byte (G1). The legal range for the 4-bit field is between 0000 and 1000, representing zero to eight errors. Any other value is interpreted as zero errors.

**Far End Receive Failure (FERF)** - a line error asserted when a 110 binary pattern is detected in bits 6, 7, 8 of the K2 byte for five consecutive frames. A line FERF is removed when any pattern other than 110 is detected in these bits for five consecutive frames.

**Far-End** - in a relationship between two devices in a circuit, the far-end device is the one that is remote.

**Face Contact (FC)** - Designation for fiber optic connector designed by Nippon Telegraph and Telephone which features a movable anti-rotation key allowing good repeatable performance despite numerous mating. Normally referred to as Fiber Connector, FC actually stands for Face Contact and sometimes linked with PC (Point Contact), designated as FC or FC-PC.

**FCC Part 68 -** The FCC rules regulating the direct connection of non-telephone company provided equipment to the public telephone network.

**Federal Communications Commission (FCC)** - a board of commissioners appointed by the President under the Communications Act of 1934, with the authority to regulate all interstate telecommunications originating in the United States, including transmission over phone lines.

**Fiber Distributed Data Interface (FDDI) -** high-speed data network that uses fiber-optic as the physical medium. Operates in similar manner to Ethernet or Token Ring, only faster.

**File Transfer Protocol (FTP) -** a TCP/IP protocol that lets a user on one computer access, and transfer data to and from, another computer over a network. ftp is usually the name of the program the user invokes to accomplish this task.

First-In, First-Out (FIFO) - method of coordinating the sequential flow of data through a buffer.

**Flag** - a bit pattern of six binary "1"s bounded by a binary "0" at each end (forms a 0111 1110 or Hex "7E"). It is used to mark the beginning and/or end of a frame.

**Flow Control -** The way in which information is controlled in a network to prevent loss of data when the receiving buffer is near its capacity.

**ForeThought PNNI (FT-PNNI) -** a FORE Systems routing and signalling protocol that uses private ATM (NSAP) addresses; a precursor to ATM Forum PNNI (see PNNI).

**Forward Error Correction (FEC)** - A technique used by a receiver for correcting errors incurred in transmission over a communications channel without requiring retransmission of any information by the transmitter; typically involves a convolution of the transmitted bits and the appending of extra bits by both the receiver and transmitter using a common algorithm.

**Forward Explicit Congestion Notification (FECN)** - Bit set by a Frame Relay network to inform data terminal equipment (DTE) receiving the frame that congestion was experienced in the path from source to destination. DTE receiving frames with the FECN bit set can request that higher-level protocols take flow control action as appropriate.

Fractional T1 - the use of bandwidth in 64Kbps increments up to 1.544Mbps from a T1 facility.

**Frame -** a variable length group of data bits with a specific format containing flags at the beginning and end to provide demarcation.

**Frame Check Sequence (FCS) -** In bit-oriented protocols, a 16-bit field that contains transmission error checking information, usually appended to the end of the frame.

**Frame Relay -** a fast packet switching protocol based on the LAPD protocol of ISDN that performs routing and transfer with less overhead processing than X.25.

**Frame Synchronization Error** - an error in which one or more time slot framing bits are in error.

**Frame-Based UNI (FUNI) -** An ATM switch-based interface which accepts frame-based ATM traffic and converts it into cells.

**Frame-Relay Service (FRS) -** A connection oriented service that is capable of carrying up to 4096 bytes per frame.

**Framing -** a protocol that separates incoming bits into identifiable groups so that the receiving multiplexer recognizes the grouping.

**Frequency Division Multiplexing (FDM) -** a method of dividing an available frequency range into parts with each having enough bandwidth to carry one channel.

**Gbps** - gigabits per second (billion)

**Generic Cell Rate Algorithm (GCRA) -** an algorithm which is employed in traffic policing and is part of the user/network service contract. The GCRA is a scheduling algorithm which ensures that cells are marked as conforming when they arrive when expected or later than expected and non-conforming when they arrive sooner than expected.

**Generic Connection Admission Control (GCAC) -** This is a process to determine if a link has potentially enough resources to support a connection.

Generic Flow Control (GFC) - the first four bits of the first byte in an ATM cell header. Used to control the flow of traffic across the User-to-Network Interface (UNI), and thus into the network. Exact mechanisms for flow control are still under investigation and no explicit definition for this field exists at this time. (This field is used only at the UNI; for NNI-NNI use (between network nodes), these four bits provide additional network address capacity, and are appended to the VPI field.)

**GIO** - a proprietary bus architecture used in certain Silicon Graphics, Inc. workstations.

**Header -** protocol control information located at the beginning of a protocol data unit.

**Header Error Control (HEC) -** a CRC code located in the last byte of an ATM cell header that is used for checking cell header integrity only.

**High Density Bipolar (HDB3) -** A bipolar coding method that does not allow more than 3 consecutive zeroes.

**High Level Data Link Control (HDLC) -** An ITU-TSS link layer protocol standard for point-to-point and multi-point communications.

**High Performance Parallel Interface (HIPPI) -** ANSI standard that extends the computer bus over fairly short distances at speeds of 800 and 1600 Mbps.

**High-Speed Serial Interface (HSSI) -** a serial communications connection that operates at speeds of up to 1.544 Mbps.

Host - In a network, the primary or controlling computer in a multiple computer installation.

**HPUX -** the Hewlett-Packard version of UNIX.

**Hub** - a device that connects several other devices, usually in a star topology.

**I/O Module -** FORE's interface cards for the LAX-20 LAN Access Switch, designed to connect Ethernet, Token Ring, and FDDI LANs to *ForeRunner* ATM networks.

**Institute of Electrical and Electronics Engineers (IEEE) -** the world's largest technical professional society. Based in the U.S., the IEEE sponsors technical conferences, symposia & local meetings worldwide, publishes nearly 25% of the world's technical papers in electrical, electronics & computer engineering, provides educational programs for members, and promotes standardization.

**IEEE 802 -** Standards for the interconnection of LAN computer equipment. Deals with the Data Link Layers of the ISO Reference Model for OSI.

**IEEE 802.1 -** Defines the high-level network interfaces such as architecture, internetworking and network management.

**IEEE 802.2 -** Defines the Logical Link Control interface between the Data Link and Network Layers.

**IEEE 802.3 -** Defines CSMA/CD (Ethernet).

**IEEE 802.4 -** Defines the token-passing bus.

**IEEE 802.5 -** Defines the Token Ring access methodology. This standard incorporates IBM's Token Ring specifications.

**IEEE 802.6 -** Defines Metropolitan Area Networks.

**IEEE 802.7 -** The broadband technical advisory group.

**Integrated Services Digital Network (ISDN) -** an emerging technology that is beginning to be offered by the telephone carriers of the world. ISDN combines voice and digital network services into a single medium or wire.

**Interexchange Carriers (IXC) -** Long-distance communications companies that provide service between Local Access Transport Areas (LATAs).

**Interface Data -** the unit of information transferred to/from the upper layer in a single interaction across a SAP. Each Interface Data Unit (IDU) controls interface information and may also contain the whole or part of the SDU.

Interface Data Unit (IDU) - The unit of information transferred to/from the upper layer in a single interaction across the SAP. Each IDU contains interface control information and may also contain the whole or part of the SDU.

**Interim Local Management Interface (ILMI) -** the standard that specifies the use of the Simple Network Management Protocol (SNMP) and an ATM management information base (MIB) to provide network status and configuration information.

**Intermediate System (IS)** - a system that provides forwarding functions or relaying functions or both for a specific ATM connection. OAM cells may be generated and received.

**International Standards Organization (ISO)** - a voluntary, non treaty organization founded in 1946 that is responsible for creating international standards in many areas, including computers and communications.

**International Telephone and Telegraph Consultative Committee (CCITT) -** the international standards body for telecommunications.

**Internet -** (note the capital "I") the largest internet in the world including large national backbone nets and many regional and local networks worldwide. The Internet uses the TCP/IP suite. Networks with only e-mail connectivity are not considered on the Internet.

**internet -** while an internet is a network, the term "internet" is usually used to refer to a collection of networks interconnected with routers.

Internet Addresses - the numbers used to identify hosts on an internet network. Internet host numbers are divided into two parts; the first is the network number and the second, or local, part is a host number on that particular network. There are also three classes of networks in the Internet, based on the number of hosts on a given network. Large networks are classified as Class A, having addresses in the range 1-126 and having a maximum of 16,387,064 hosts. Medium networks are classified as Class B, with addresses in the range 128-191 and with a maximum of 64,516 hosts. Small networks are classified as Class C, having addresses in the range 192-254 with a maximum of 254 hosts. Addresses are given as dotted decimal numbers in the following format:

#### nnn.nnn.nnn.nnn

In a Class A network, the first of the numbers is the network number, the last three numbers are the local host address.

In a Class B network, the first two numbers are the network, the last two are the local host address.

In a Class C network, the first three numbers are the network address, the last number is the local host address.

The following table summarizes the classes and sizes:

Class	First #	Max# Hosts
Α	1-126	16,387,064
В	129-191	64,516
C	192-223	254

## Glossary

Network mask values are used to identify the network portion and the host portion of the address. Default network masks are as follows:

Class A - 255.0.0.0

Class B - 255.255.0.0

Class C - 255,255,255.0

Subnet masking is used when a portion of the host ID is used to identify a subnetwork. For example, if a portion of a Class B network address is used for a subnetwork, the mask could be set as 255.255.255.0. This would allow the third byte to be used as a subnetwork address. All hosts on the network would still use the IP address to get on the Internet.

**Internet Control Message Protocol (ICMP)** - the protocol that handles errors and control messages at the IP layer. ICMP is actually a part of the IP protocol layer. It can generate error messages, test packets, and informational messages related to IP.

**Internet Engineering Task Force (IETF) -** a large, open, international community of network designers, operators, vendors and researchers whose purpose is to coordinate the operation, management and evolution of the Internet to resolve short- and mid-range protocol and architectural issues.

**Internet Protocol (IP) -** a connectionless, best-effort packet switching protocol that offers a common layer over dissimilar networks.

**Internetwork Packet Exchange (IPX) Protocol -** a NetWare protocol similar to the Xerox Network Systems (XNS) protocol that provides datagram delivery of messages.

**Interoperability -** The ability of software and hardware on multiple machines, from multiple vendors, to communicate.

**Interworking Function (IWF) -** provides a means for two different technologies to interoperate.

**IP Address -** a unique 32-bit integer used to identify a device in an IP network. You will most commonly see IP addresses written in "dot" notation (e.g., 192.228.32.14).

**IP Netmask -** a 32-bit pattern that is combined with an IP address to determine which bits of an IP address denote the network number and which denote the host number. Netmasks are useful for sub-dividing IP networks. IP netmasks are written in "dot" notation (e.g., 255.255.0.0).

**ISA Bus -** a bus standard developed by IBM for expansion cards in the first IBM PC. The original bus supported a data path only 8 bits wide. IBM subsequently developed a 16-bit version for its AT class computers. The 16-bit AT ISA bus supports both 8- and 16-bit cards. The 8-bit bus is commonly called the PC/XT bus, and the 16-bit bus is called the AT bus.

**Isochronous -** signals carrying embedded timing information or signals that are dependent on uniform timing; usually associated with voice and/or video transmission.

International Telecommunications Union Telecommunications (ITU-T) - an international body of member countries whose task is to define recommendations and standards relating to the international telecommunications industry. The fundamental standards for ATM have been defined and published by the ITU-T (Previously CCITT).

**J2 -** Wide-area digital transmission scheme used predominantly in Japan that carries data at a rate of 6.312 Mbps.

**Jitter -** analog communication line distortion caused by variations of a signal from its reference timing position.

**Joint Photographic Experts Group (JPEG) -** An ISO Standards group that defines how to compress still pictures.

**Jumper -** a patch cable or wire used to establish a circuit, often temporarily, for testing or diagnostics; also, the devices, shorting blocks, used to connect adjacent exposed pins on a printed circuit board that control the functionality of the card.

**Kbps** - kilobits per second (thousand)

**LAN Access Concentrator -** a LAN access device that allows a shared transmission medium to accommodate more data sources than there are channels currently available within the transmission medium.

**LAN Emulation Address Resolution Protocol (LE\_ARP) -** A message issued by a LE client to solicit the ATM address of another function.

**LAN Emulation Client (LEC) -** the component in an end system that performs data forwarding, address resolution, and other control functions when communicating with other components within an ELAN.

**LAN Emulation Configuration Server (LECS) -** the LECS is responsible for the initial configuration of LECs. It provides information about available ELANs that a LEC may join, together with the addresses of the LES and BUS associated with each ELAN.

**LAN Emulation Server (LES) -** the LES implements the control coordination function for an ELAN by registering and resolving MAC addresses to ATM addresses.

**LAN Emulation (LANE) -** technology that allows an ATM network to function as a LAN backbone. The ATM network must provide multicast and broadcast support, address mapping (MAC-to-ATM), SVC management, and a usable packet format. LANE also defines Ethernet and Token Ring ELANs.

lane - a program that provides control over the execution of the LAN Emulation Server (LES), Broadcast/Unknown Server (BUS), and LAN Emulation Configuration Server (LECS) on the local host.

**Latency -** The time interval between a network station seeking access to a transmission channel and that access being granted or received.

**Layer Entity -** an active layer within an element.

**Layer Function -** a part of the activity of the layer entities.

**Layer Service -** a capability of a layer and the layers beneath it that is provided to the upper layer entities at the boundary between that layer and the next higher layer.

**Layer User Data -** the information transferred between corresponding entities on behalf of the upper layer or layer management entities for which they are providing services.

**le -** a FORE program that implements both the LAN Emulation Server (LES) and the Broadcast/Unknown Server (BUS).

**Leaky Bucket -** informal cell policing term for the Generic Cell Rate Algorithm which in effect receives cells into a bucket and leaks them out at the specified or contracted rate (i.e., PCR).

**Least Significant Bit (LSB) -** lowest order bit in the binary representation of a numerical value.

**lecs -** a FORE program that implements the assignment of individual LECs to different emulated LANs.

**leq -** a FORE program that provides information about an ELAN. This information is obtained from the LES, and includes MAC addresses registered on the ELAN together with their corresponding ATM addresses.

**Line Build Out (LBO)** - Because T1 circuits require the last span to lose 15-22.5 dB, a selectable output attenuation is generally required of DTE equipment (typical selections include 0.0, 7.5 and 15 dB of loss at 772 KHz).

**Line Code Violations (LCV) -** Error Event. A Line Coding Violation (LCV) is the occurrence of either a Bipolar Violation (BPV) or Excessive Zeroes (EXZ) Error Event.

**Link -** An entity that defines a topological relationship (including available transport capacity) between two nodes in different subnetworks. Multiple links may exist between a pair of subnetworks. Synonymous with logical link.

**Link Access Procedure, Balanced (LAPB) -** Data link protocol in the X.25 protocol stack. LAPB is a bit-oriented protocol derived from HDLC. See also HDLC and X.25.

**Link Down Trap -** an SNMP trap, sent when an interface changes from a normal state to an error state, or is disconnected.

Link Layer - layer in the OSI model regarding transmission of data between network nodes.

**Link Up Trap -** an SNMP trap, sent when an interface changes from an error condition to a normal state.

**Load Sharing -** Two or more computers in a system that share the load during peak hours. During periods of non peak hours, one computer can manage the entire load with the other acting as a backup.

**Local Access and Transport Area (LATA) -** Geographic boundaries of the local telephone network, specified by the FCC, in which a single LEC may perform its operations. Communications outside or between LATAs are provided by IXCs.

**Local Area Network (LAN) -** a data network intended to serve an area of only a few square kilometers or less. Because the network is known to cover only a small area, optimizations can be made in the network signal protocols that permit higher data rates.

**Logical Link Control (LLC) -** protocol developed by the IEEE 802 committee for data-link-layer transmission control; the upper sublayer of the IEEE Layer 2 (OSI) protocol that complements the MAC protocol; IEEE standard 802.2; includes end-system addressing and error checking.

**Loopback** - a troubleshooting technique that returns a transmitted signal to its source so that the signal can be analyzed for errors. Typically, a loopback is set at various points in a line until the section of the line that is causing the problem is discovered.

**looptest -** program that tests an interface for basic cell reception and transmission functionality, usually used for diagnostic purposes to determine if an interface is functioning properly.

Loss Of Frame (LOF) - a type of transmission error that may occur in wide-area carrier lines.

**Loss Of Pointer (LOP) -** a type of transmission error that may occur in wide-area carrier lines.

**Loss Of Signal (LOS) -** a type of transmission error that may occur in wide-area carrier lines, or a condition declared when the DTE senses a loss of a DS1 signal from the CPE for more the 150 milliseconds (the DTE generally responds with an all ones "Blue or AIS" signal).

**Management Information Base (MIB) -** the set of parameters that an SNMP management station can query or set in the SNMP agent of a networked device (e.g., router).

**Maximum Burst Size (MBS)** - the Burst Tolerance (BT) is conveyed through the MBS which is coded as a number of cells. The BT together with the SCR and the GCRA determine the MBS that may be transmitted at the peak rate and still be in conformance with the GCRA.

**Maximum Burst Tolerance -** the largest burst of data that a network device is guaranteed to handle without discarding cells or packets. Bursts of data larger than the maximum burst size may be subject to discard.

**Maximum Cell Delay Variance (MCDV) -** This is the maximum two-point CDV objective across a link or node for the specified service category.

**Maximum Cell Loss Ratio (MCLR) -** This is the maximum ratio of the number of cells that do not make it across the link or node to the total number of cells arriving at the link or node.

**Maximum Cell Transfer Delay (MCTD) -** This is the sum of the fixed delay component across the link or node and MCDV.

**Maximum Transmission Unit (MTU) -** the largest unit of data that can be sent over a type of physical medium.

Mbps - megabits per second (million)

**Media Access Control (MAC) -** a media-specific access control protocol within IEEE 802 specifications; currently includes variations for Token Ring, token bus, and CSMA/CD; the lower sublayer of the IEEE's link layer (OSI), which complements the Logical Link Control (LLC).

**Media Attachment Unit (MAU)** - device used in Ethernet and IEEE 802.3 networks that provides the interface between the AUI port of a station and the common medium of the Ethernet. The MAU, which can be built into a station or can be a separate device, performs physical layer functions including conversion of the digital data from the Ethernet interface, collision detection, and injection of bits onto the network.

Media Interface Connector (MIC) - fiber optic connector that joins fiber to the FDDI controller.

**Message Identifier (MID) -** message identifier used to associate ATM cells that carry segments from the same higher layer packet.

**Metasignalling -** an ATM Layer Management (LM) process that manages different types of signalling and possibly semipermanent virtual channels (VCs), including the assignment, removal, and checking of VCs.

**Metasignalling VCs -** the standardized VCs that convey metasignalling information across a User-to-Network Interface (UNI).

**Metropolitan Area Network (MAN) -** network designed to carry data over an area larger than a campus such as an entire city and its outlying area.

**MicroChannel -** a proprietary 16- or 32-bit bus developed by IBM for its PS/2 computers' internal expansion cards; also offered by others.

**Minimum Cell Rate (MCR) -** parameter defined by the ATM Forum for ATM traffic management, defined only for ABR transmissions and specifying the minimum value for the ACR.

Most Significant Bit (MSB) - highest order bit in the binary representation of a numerical value.

**Motion Picture Experts Group (MPEG) -** ISO group dealing with video and audio compression techniques and mechanisms for multiplexing and synchronizing various media streams.

**MPOA Client -** A device which implements the client side of one or more of the MPOA protocols, (i.e., is a SCP client and/or an RDP client. An MPOA Client is either an Edge Device Functional Group (EDFG) or a Host Behavior Functional Group (HBFG).

**MPOA Server -** An MPOA Server is any one of an ICFG or RSFG.

**MPOA Service Area -** The collection of server functions and their clients. A collection of physical devices consisting of an MPOA server plus the set of clients served by that server.

**MPOA Target -** A set of protocol address, path attributes, (e.g., internetwork layer QoS, other information derivable from received packet) describing the intended destination and its path attributes that MPOA devices may use as lookup keys.

**Mu-Law -** The PCM coding and companding standard used in Japan and North America.

**Multicasting -** The ability to broadcast messages to one node or a select group of nodes.

Multi-homed - a device having both an ATM and another network connection, like Ethernet.

**Multimode Fiber Optic Cable (MMF)** - fiber optic cable in which the signal or light propagates in multiple modes or paths. Since these paths may have varying lengths, a transmitted pulse of light may be received at different times and smeared to the point that pulses may interfere with surrounding pulses. This may cause the signal to be difficult or impossible to receive. This pulse dispersion sometimes limits the distance over which a MMF link can operate.

**Multiplexing -** a function within a layer that interleaves the information from multiple connections into one connection (see demultiplexing).

**Multipoint Access -** user access in which more than one terminal equipment (TE) is supported by a single network termination.

**Multipoint-to-Multipoint Connection -** a collection of associated ATM VC or VP links, and their associated endpoint nodes, with the following properties:

- 1. All N nodes in the connection, called Endpoints, serve as a Root Node in a Point-to-Multipoint connection to all of the (N-1) remaining endpoints.
- 2. Each of the endpoints can send information directly to any other endpoint, but the receiving endpoint cannot distinguish which of the endpoints is sending information without additional (e.g., higher layer) information.

**Multipoint-to-Point Connection -** a Point-to-Multipoint Connection may have zero bandwidth from the Root Node to the Leaf Nodes, and non-zero return bandwidth from the Leaf Nodes to the Root Node. Such a connection is also known as a Multipoint-to-Point Connection.

**Multiprotocol over ATM (MPOA) -** An effort taking place in the ATM Forum to standardize protocols for the purpose of running multiple network layer protocols over ATM.

Narrowband Channel - sub-voicegrade channel with a speed range of 100 to 200 bps.

National TV Standards Committee (NTSC) - Started in the US in 1953 from a specification laid down by the National Television Standards Committee. It takes the B-Y and R-Y color difference signals, attenuates them to I and Q, then modulates them using double-sideband suppressed subcarrier at 3.58MHz. The carrier reference is sent to the receiver as a burst during the back porch. An industry group that defines how television signals are encoded and transmitted in the US. (See also PAL, SECAM for non-U.S. countries).

**Near-End -** in a relationship between two devices in a circuit, the near-end device is the one that is local.

**Network Layer** - Layer three In the OSI model, the layer that is responsible for routing data across the network.

**Network Management Entity (NM) -** body of software in a switching system that provides the ability to manage the PNNI protocol. NM interacts with the PNNI protocol through the MIB.

**Network Management Layer (NML) -** an abstraction of the functions provided by systems which manage network elements on a collective basis, providing end-to-end network monitoring.

**Network Management Station (NMS) -** system responsible for managing a network or portion of a network by talking to network management agents, which reside in the managed nodes.

**Network Module -** ATM port interface cards which may be individually added to or removed from any *ForeRunner* ATM switch to provide a diverse choice of connection alternatives.

**Network Parameter Control (NPC)** - Defined as the set of actions taken by the network to monitor and control traffic from the NNI. Its main purpose is to protect network resources from malicious as well as unintentional misbehavior which can affect the QoS of other already established connections by detecting violations of negotiated parameters and taking appropriate actions. Refer to UPC.

**Network Redundancy -** Duplicated network equipment and/or data which can provide a backup in case of network failures.

**Network Service Access Point (NSAP) -** OSI generic standard for a network address consisting of 20 octets. ATM has specified E.164 for public network addressing and the NSAP address structure for private network addresses.

**Network-to-Network Interface or Network Node Interface (NNI) -** the interface between two public network pieces of equipment.

**Node -** A computer or other device when considered as part of a network.

**Non Return to Zero (NRZ) -** a binary encoding scheme in which ones and zeroes are represented by opposite and alternating high and low voltages and where there is no return to a zero (reference) voltage between encoded bits.

**Non Return to Zero Inverted (NRZI) -** A binary encoding scheme that inverts the signal on a "1" and leaves the signal unchanged for a "0". (Also called transition encoding.)

Nonvolatile Storage - Memory storage that does not lose its contents when power is turned off.

**NuBus -** a high-speed bus used in Macintosh computers, structured so users can put a card into any slot on the board without creating conflict over the priority between those cards.

nx64K - This refers to a circuit bandwidth or speed provided by the aggregation of nx64 kbps channels (where n= integer > 1). The 64K or DS0 channel is the basic rate provided by the T Carrier systems.

**Nyquist Theorem** - In communications theory, a formula stating that two samples per cycle is sufficient to characterize a bandwidth limited analog signal; in other words, the sampling rate must be twice the highest frequency component of the signal (i.e., sample 4 KHz analog voice channels 8000 times per second).

**Object Identifier (OID) -** the address of a MIB variable.

Octet - a grouping of 8 bits; similar, but not identical to, a byte.

One's Density - The requirement for digital transmission lines in the public switched telephone network that eight consecutive "0"s cannot be in a digital data stream; exists because repeaters and clocking devices within the network will lose timing after receiving eight "0"s in a row; a number of techniques are used to insert a "1" after every seventh-consecutive "0" (see Bit Stuffing).

**Open Shortest Path First (OSPF) Protocol -** a routing algorithm for IP that incorporates least-cost, equal-cost, and load balancing.

**Open Systems Interconnection (OSI) -** the 7-layer suite of protocols designed by ISO committees to be the international standard computer network architecture.

**OpenView -** Hewlett-Packard's network management software.

**Operation and Maintenance (OAM) Cell -** a cell that contains ATM LM information. It does not form part of the upper layer information transfer.

**Optical Carrier level-n (OC-n) -** The optical counterpart of STS-n (the basic rate of 51.84 Mbps on which SONET is based is referred to as OC-1 or STS-1).

**Organizationally Unique Identifier (OUI) -** Part of RFC 1483. A three-octet field in the SubNetwork Attachment Point (SNAP) header, identifying an organization which administers the meaning of the following two octet Protocol Identifier (PID) field in the SNAP header. Together they identify a distinct routed or bridged protocol.

**Out-of-Band Management -** refers to switch configuration via the serial port or over Ethernet, not ATM.

Out-of-Frame (OOF) - a signal condition and alarm in which some or all framing bits are lost.

**Packet -** An arbitrary collection of data grouped and transmitted with its user identification over a shared facility.

**Packet Assembler Disassembler (PAD) -** interface device that buffers data sent to/from character mode devices, and assembles and disassembles the packets needed for X.25 operation.

**Packet Internet Groper (ping) -** a program used to test reachability of destinations by sending them an ICMP echo request and waiting for a reply.

**Packet Level Protocol (PLP) -** Network layer protocol in the X.25 protocol stack. Sometimes called X.25 Level 3 or X.25 Protocol.

**Packet Switched Network (PSN) -** a network designed to carry data in the form of packets. The packet and its format is internal to that network.

**Packet Switching -** a communications paradigm in which packets (messages) are individually routed between hosts with no previously established communications path.

**Payload Scrambling -** a technique that eliminates certain bit patterns that may occur within an ATM cell payload that could be misinterpreted by certain sensitive transmission equipment as an alarm condition.

**Payload Type (PT)** - bits 2...4 in the fourth byte of an ATM cell header. The PT indicates the type of information carried by the cell. At this time, values 0...3 are used to identify various types of user data, values 4 and 5 indicate management information, and values 6 and 7 are reserved for future use.

**Peak Cell Rate -** at the PHY Layer SAP of a point-to-point VCC, the Peak Cell Rate is the inverse of the minimum inter-arrival time T0 of the request to send an ATM-SDU.

**Peak Cell Rate (PCR) -** parameter defined by the ATM Forum for ATM traffic management. In CBR transmissions, PCR determines how often data samples are sent. In ABR transmissions, PCR determines the maximum value of the ACR.

**Peer Entities -** entities within the same layer.

**Peripheral Component Interconnect (PCI) -** a local-bus standard created by Intel.

Permanent Virtual Channel Connection (PVCC) - A Virtual Channel Connection (VCC) is an ATM connection where switching is performed on the VPI/VCI fields of each cell. A Permanent VCC is one which is provisioned through some network management function and left up indefinitely.

**Permanent Virtual Circuit (or Channel) (PVC) -** a circuit or channel through an ATM network provisioned by a carrier between two endpoints; used for dedicated long-term information transport between locations.

**Permanent Virtual Path Connection (PVPC) -** A Virtual Path Connection (VPC) is an ATM connection where switching is performed on the VPI field only of each cell. A PVPC is one which is provisioned through some network management function and left up indefinitely.

Phase Alternate Line (PAL) - Largely a German/British development in the late 60s, used in the UK and much of Europe. The B-Y and R-Y signals are weighted to U and V, then modulated onto a double-sideband suppressed subcarrier at 4.43MHz. The V (R-Y) signal's phase is turned through 180 degrees on each alternate line. This gets rid of NTSC's hue changes with phase errors at the expense of de-saturation. The carrier reference is sent as a burst in the back porch. The phase of the burst is alternated every line to convey the phase switching of the V signal. The burst's average phase is -V. (see NTSC for U.S.).

**Physical Layer (PHY) -** the actual cards, wires, and/or fiber-optic cabling used to connect computers, routers, and switches.

**Physical Layer Connection -** an association established by the PHY between two or more ATM-entities. A PHY connection consists of the concatenation of PHY links in order to provide an end-to-end transfer capability to PHY SAPs.

**Physical Layer Convergence Protocol (PLCP) -** a framing protocol that runs on top of the T1 or E1 framing protocol.

**Physical Medium (PM)** - Refers to the actual physical interfaces. Several interfaces are defined including STS-1, STS-3c, STS-12c, STM-1, STM-4, DS1, E1, DS2, E3, DS3, E4, FDDI-based, Fiber Channel-based, and STP. These range in speeds from 1.544Mbps through 622.08 Mbps.

**Physical Medium Dependent (PMD) -** a sublayer concerned with the bit transfer between two network nodes. It deals with wave shapes, timing recovery, line coding, and electro-optic conversions for fiber based links.

**Plesiochronous -** two signals are plesiochronous if their corresponding significant instants occur at nominally the same rate, with variations in rate constrained to specified limits.

**PMD** - in HP OpenView, the background process that receives and forwards events.

**Point of Demarcation -** the dividing line between a carrier and the customer premise that is governed by strict standards that define the characteristics of the equipment on each side of the demarcation. Equipment on one side of the point of demarcation is the responsibility of the customer. Equipment on the other side of the point of demarcation is the responsibility of the carrier.

**Point-to-Multipoint Connection -** a collection of associated ATM VC or VP links, with associated endpoint nodes, with the following properties:

- 1. One ATM link, called the Root Link, serves as the root in a simple tree topology. When the Root node sends information, all of the remaining nodes on the connection, called Leaf nodes, receive copies of the information.
- 2. Each of the Leaf Nodes on the connection can send information directly to the Root Node. The Root Node cannot distinguish which Leaf is sending information without additional (higher layer) information. (See the following note for Phase 1.)
- 3. The Leaf Nodes cannot communicate directly to each other with this connection type.

Note: Phase 1 signalling does not support traffic sent from a Leaf to the Root.

**Point-to-Point Connection -** a connection with only two endpoints.

**Point-to-Point Protocol (PPP) -** Provides a method for transmitting packets over serial point-to-point links.

**Policing -** the function that ensures that a network device does not accept traffic that exceeds the configured bandwidth of a connection.

**Port Identifier -** The identifier assigned by a logical node to represent the point of attachment of a link to that node.

Presentation Layer - Sixth layer of the OSI model, providing services to the application layer.

**Primary Reference Source (PRS) -** Equipment that provides a timing signal whose long-term accuracy is maintained at  $1 \times 10$  -11 or better with verification to universal coordinated time (UTC) and whose timing signal is used as the basis of reference for the control of other clocks within a network.

**Primitive -** an abstract, implementation-independent interaction between a layer service user and a layer service provider.

**Priority -** the parameter of ATM connections that determines the order in which they are reduced from the peak cell rate to the sustained cell rate in times of congestion. Connections with lower priority (4 is low, 1 is high) are reduced first.

**Private Branch Exchange (PBX) -** a private phone system (switch) that connects to the public telephone network and offers in-house connectivity. To reach an outside line, the user must dial a digit like 8 or 9.

**Private Network Node Interface or Private Network-to-Network Interface (PNNI) -** a protocol that defines the interaction of private ATM switches or groups of private ATM switches

**Programmable Read-Only Memory (PROM) -** a chip-based information storage area that can be recorded by an operator but erased only through a physical process.

**Protocol -** a set of rules and formats (semantic and syntactic) that determines the communication behavior of layer entities in the performance of the layer functions.

**Protocol Control Information -** the information exchanged between corresponding entities using a lower layer connection to coordinate their joint operation.

**Protocol Data Unit (PDU) -** a unit of data specified in a layer protocol and consisting of protocol control information and layer user data.

**Proxy** - the process in which one system acts for another system to answer protocol requests.

**Proxy Agent -** an agent that queries on behalf of the manager, used to monitor objects that are not directly manageable.

**Public Data Network (PDN) -** a network designed primarily for data transmission and intended for sharing by many users from many organizations.

**Pulse Code Modulation (PCM) -** a modulation scheme that samples the information signals and transmits a series of coded pulses to represent the data.

**Q.2931 -** Derived from Q.93B, the narrowband ISDN signalling protocol, an ITU standard describing the signalling protocol to be used by switched virtual circuits on ATM LANs.

**Quality of Service (QoS) -** Quality of Service is defined on an end-to-end basis in terms of the following attributes of the end-to-end ATM connection:

Cell Loss Ratio

Cell Transfer Delay

Cell Delay Variation

**Queuing Delay (QD)** - refers to the delay imposed on a cell by its having to be buffered because of unavailability of resources to pass the cell onto the next network function or element. This buffering could be a result of oversubscription of a physical link, or due to a connection of higher priority or tighter service constraints getting the resource of the physical link.

Radio Frequency Interference (RFI) - the unintentional transmission of radio signals. Computer equipment and wiring can both generate and receive RFI.

**Real-Time Clock -** a clock that maintains the time of day, in contrast to a clock that is used to time the electrical pulses on a circuit.

**Red Alarm -** In T1, a red alarm is generated for a locally detected failure such as when a condition like OOF exists for 2.5 seconds, causing a CGA, (Carrier Group Alarm).

**Reduced Instruction Set Computer (RISC) -** a generic name for CPUs that use a simpler instruction set than more traditional designs.

**Redundancy -** In a data transmission, the fragments of characters and bits that can be eliminated with no loss of information.

**Registration -** The address registration function is the mechanism by which Clients provide address information to the LAN Emulation Server.

**Relaying -** a function of a layer by means of which a layer entity receives data from a corresponding entity and transmits it to another corresponding entity.

**Request To Send (RTS) -** an RS-232 modem interface signal (sent from the DTE to the modem on pin 4) which indicates that the DTE has data to transmit.

**Requests For Comment (RFCs) -** IETF documents suggesting protocols and policies of the Internet, inviting comments as to the quality and validity of those policies. These comments are collected and analyzed by the IETF in order to finalize Internet standards.

RFC1483 - Multiprotocol Encapsulation over ATM Adaptation Layer 5.

RFC1490 - Multiprotocol Interconnect over Frame Relay.

**RFC1577 -** Classical IP and ARP over ATM.

**RFC1755 -** ATM Signaling Support for IP over ATM.

**Robbed-Bit Signaling -** In T1, refers to the use of the least significant bit of every word of frames 6 and 12 (D4), or 6, 12, 18, and 24 (ESF) for signaling purposes.

**Route Server -** A physical device that runs one or more network layer routing protocols, and which uses a route query protocol in order to provide network layer routing forwarding descriptions to clients.

**Router -** a device that forwards traffic between networks or subnetworks based on network layer information.

**Routing Domain (RD) -** A group of topologically contiguous systems which are running one instance of routing.

**Routing Information Protocol (RIP) -** a distance vector-based protocol that provides a measure of distance, or hops, from a transmitting workstation to a receiving workstation.

**Routing Protocol** - A general term indicating a protocol run between routers and/or route servers in order to exchange information used to allow computation of routes. The result of the routing computation will be one or more forwarding descriptions.

**SBus -** hardware interface for add-in boards in later-version Sun 3 workstations.

**Scalable Processor Architecture Reduced instruction set Computer (SPARC) -** a powerful workstation similar to a reduced-instruction-set-computing (RISC) workstation.

**Segment -** a single ATM link or group of interconnected ATM links of an ATM connection.

Segmentation And Reassembly (SAR) - the SAR accepts PDUs from the CS and divides them into very small segments (44 bytes long). If the CS-PDU is less than 44 bytes, it is padded to 44 with zeroes. A two-byte header and trailer are added to this basic segment. The header identifies the message type (beginning, end, continuation, or single) and contains sequence numbering and message identification. The trailer gives the SAR-PDU payload length, exclusive of pad, and contains a CRC check to ensure the SAR-PDU integrity. The result is a 48-byte PDU that fits into the payload field of an ATM cell.

**Selector (SEL) -** A subfield carried in SETUP message part of ATM endpoint address Domain specific Part (DSP) defined by ISO 10589, not used for ATM network routing, used by ATM end systems only.

**Semipermanent Connection -** a connection established via a service order or via network management.

**Serial Line IP (SLIP) -** A protocol used to run IP over serial lines, such as telephone circuits or RS-232 cables, interconnecting two systems.

**Service Access Point (SAP) -** the point at which an entity of a layer provides services to its LM entity or to an entity of the next higher layer.

**Service Data Unit (SDU) -** a unit of interface information whose identity is preserved from one end of a layer connection to the other.

**Service Specific Connection Oriented Protocol (SSCOP) -** an adaptation layer protocol defined in ITU-T Specification: Q.2110.

**Service Specific Convergence Sublayer (SSCS) -** The portion of the convergence sublayer that is dependent upon the type of traffic that is being converted.

**Session Layer -** Layer 5 in the OSI model that is responsible for establishing and managing sessions between the application programs running in different nodes.

**Severely Errored Seconds (SES) -** a second during which more event errors have occurred than the SES threshold (normally 10-3).

**Shaping Descriptor -** *n* ordered pairs of GCRA parameters (I,L) used to define the negotiated traffic shape of an APP connection. The traffic shape refers to the load-balancing of a network, where load-balancing means configuring data flows to maximize network efficiency.

**Shielded Pair -** Two insulated wires in a cable wrapped with metallic braid or foil to prevent interference and provide noise free transmission.

**Shielded Twisted Pair (STP) -** two or more insulated wires, twisted together and then wrapped in a cable with metallic braid or foil to prevent interference and offer noise-free transmissions.

**Signaling System No. 7 (SS7) -** The SS7 protocol has been specified by ITU-T and is a protocol for interexchange signaling.

Simple and Efficient Adaptation Layer (SEAL) - also called AAL 5, this ATM adaptation layer assumes that higher layer processes will provide error recovery, thereby simplifying the SAR portion of the adaptation layer. Using this AAL type packs all 48 bytes of an ATM cell information field with data. It also assumes that only one message is crossing the UNI at a time. That is, multiple end-users at one location cannot interleave messages on the same VC, but must queue them for sequential transmission.

Simple Gateway Management Protocol (SGMP) - the predecessor to SNMP.

Simple Mail Transfer Protocol (SMTP) - the Internet electronic mail protocol used to transfer electronic mail between hosts.

**Simple Network Management Protocol (SNMP) -** the Internet standard protocol for managing nodes on an IP network.

**Simple Protocol for ATM Network Signalling (SPANS) -** FORE Systems' proprietary signalling protocol used for establishing SVCs between FORE Systems equipment.

**Single Mode Fiber (SMF)** - Fiber optic cable in which the signal or light propagates in a single mode or path. Since all light follows the same path or travels the same distance, a transmitted pulse is not dispersed and does not interfere with adjacent pulses. SMF fibers can support longer distances and are limited mainly by the amount of attenuation. Refer to MMF.

**Small Computer Systems Interface (SCSI) -** a standard for a controller bus that connects hardware devices to their controllers on a computer bus, typically used in small systems.

**Smart PVC (SPVC)** - a generic term for any communications medium which is permanently provisioned at the end points, but switched in the middle. In ATM, there are two kinds of SPVCs: smart permanent virtual path connections (SPVPCs) and smart permanent virtual channel connections (SPVCCs).

**snmpd** - an SMNP agent for a given adapter card.

**Source -** Part of communications system which transmits information.

**Source Address (SA) -** The address from which the message or data originated.

**Source MAC Address (SA) -** A six octet value uniquely identifying an end point and which is sent in an IEEE LAN frame header to indicate source of frame.

**Source Traffic Descriptor -** a set of traffic parameters belonging to the ATM Traffic Descriptor used during the connection set-up to capture the intrinsic traffic characteristics of the connection requested by the source.

**Spanning Tree Protocol -** provides loop-free topology in a network environment where there are redundant paths.

Static Route - a route that is entered manually into the routing table.

**Statistical Multiplexing -** a technique for allowing multiple channels and paths to share the same link, typified by the ability to give the bandwidth of a temporarily idle channel to another channel.

**Stick and Click (SC) -** Designation for an Optical Connector featuring a 2.5 mm physically contacting ferrule with a push-pull mating design. Commonly referred to as Structured Cabling, Structured Connectors or Stick and Click

**Stick and Turn (ST) -** A fiber-optic connector designed by AT&T which uses the bayonet style coupling rather than screw-on as the SMA uses. The ST is generally considered the eventual replacement for the SMA type connector.

**Store-and-Forward -** the technique of receiving a message, storing it until the proper outgoing line is available, then retransmitting it, with no direct connection between incoming and outgoing lines.

Straight Tip (ST) - see Stick and Turn.

Structured Cabling (SC) - see Stick and Click.

Structured Connectors (SC) - see Stick and Click.

**Sublayer -** a logical subdivision of a layer.

**SubNetwork Access Protocol (SNAP) -** a specially reserved variant of IEEE 802.2 encoding SNAP indicates to look further into the packet where it will fine a Type field.

**Subscriber Network Interface (SNI) -** the interface between an SMDS end user's CPE and the network directly serving the end user, supported by either a DS1 or DS3 access arrangement.

**Super Frame (SF) -** a term used to describe the repeating 12 D4 frame format that composes a standard (non-ESF) T1 service.

**Super User -** a login ID that allows unlimited access to the full range of a device's functionality, including especially the ability to reconfigure the device and set passwords.

**Sustainable Cell Rate (SCR) -** ATM Forum parameter defined for traffic management. For VBR connections, SCR determines the long-term average cell rate that can be transmitted.

**Sustained Information Rate (SIR) -** In ATM this refers to the long-term average data transmission rate across the User-to-Network Interface. In SMDS this refers to the committed information rate (similar to CIR for Frame Relay Service).

**Switch -** Equipment used to interconnect lines and trunks.

**Switched Connection -** A connection established via signaling.

**Switched Multimegabit Data Service (SMDS) -** a high-speed, datagram-based, public data network service expected to be widely used by telephone companies in their data networks.

**Switched Virtual Channel Connection (SVCC) -** A Switched VCC is one which is established and taken down dynamically through control signaling. A Virtual Channel Connection (VCC) is an ATM connection where switching is performed on the VPI/VCI fields of each cell.

**Switched Virtual Circuit (or Channel) (SVC) -** a channel established on demand by network signalling, used for information transport between two locations and lasting only for the duration of the transfer; the datacom equivalent of a dialed telephone call.

**Switched Virtual Path Connection (SVPC)** - a connection which is established and taken down dynamically through control signaling. A Virtual Path Connection (VPC) is an ATM connection where switching is performed on the VPI field only of each cell.

**Switching System -** A set of one or more systems that act together and appear as a single switch for the purposes of PNNI routing.

 $\textbf{Symmetric Connection -} a \ connection \ with \ the \ same \ bandwidth \ specified \ for \ both \ directions.$ 

**Synchronous -** signals that are sourced from the same timing reference and hence are identical in frequency.

 $\textbf{Synchronous Data Link Control (SDLC) -} \ IBM's \ data \ link \ protocol \ used \ in \ SNA \ networks.$ 

**Synchronous Optical Network (SONET) -** a body of standards that defines all aspects of transporting and managing digital traffic over optical facilities in the public network.

**Synchronous Payload Envelope (SPE) -** the payload field plus a little overhead of a basic SONET signal.

**Synchronous Transfer Mode (STM) -** a transport and switching method that depends on information occurring in regular, fixed patterns with respect to a reference such as a frame pattern.

**Synchronous Transport Signal (STS) -** a SONET electrical signal rate.

Systeme En Coleur Avec Memoire (SECAM) - Sequential and Memory Color Television - Started in France in the late 60s, and used by other countries with a political affiliation. This is. The B-Y and R-Y signals are transmitted on alternate lines modulated on an FM subcarrier. The memory is a one line delay line in the receiver to make both color difference signals available at the same time on all lines. Due to FM, the signal is robust in difficult terrain.

**Systems Network Architecture (SNA) -** a proprietary networking architecture used by IBM and IBM-compatible mainframe computers.

**T1** - a specification for a transmission line. The specification details the input and output characteristics and the bandwidth. T1 lines run at 1.544 Mbps and provide for 24 data channels. In common usage, the term "T1" is used interchangeably with "DS1."

**T1 Link -** A wideband digital carrier facility used for transmission of digitized voice, digital data, and digitized image traffic. This link is composed of two twisted-wire pairs that can carry 24 digital channels, each operating at 64K bps at the aggregate rate of 1.544M bps, full duplex. Also referred to as DS-1.

**T3** - a specification for a transmission line, the equivalent of 28 T1 lines. T3 lines run at 44.736 Mbps. In common usage, the term "T3" is used interchangeably with "DS3."

**Tachometer -** in *ForeView*, the tachometer shows the level of activity on a given port. The number in the tachometer shows the value of a chosen parameter in percentage, with a colored bar providing a semi-logarithmic representation of that percentage.

**Tagged Cell Rate (TCR) -** An ABR service parameter, TCR limits the rate at which a source may send out-of-rate forward RM-cells. TCR is a constant fixed at 10 cells/second.

**Telephony -** The conversion of voices and other sounds into electrical signals which are then transmitted by telecommunications media.

**Telnet -** a TCP/IP protocol that defines a client/server mechanism for emulating directly-connected terminal connections.

**Terminal Equipment (TE) -** Terminal equipment represents the endpoint of ATM connection(s) and termination of the various protocols within the connection(s).

**Throughput -** Measurement of the total useful information processed or communicated by a computer during a specified time period, i.e. packets per second.

**Time Division Multiplexing (TDM) -** a method of traditional digital multiplexing in which a signal occupies a fixed, repetitive time slot within a higher-rate signal.

**Token Ring -** a network access method in which the stations circulate a token. Stations with data to send must have the token to transmit their data.

**topology** - a program that displays the topology of a FORE Systems ATM network. An updated topology can be periodically re-displayed by use of the interval command option.

**Traffic -** the calls being sent and received over a communications network. Also, the packets that are sent on a data network.

**Traffic Management (TM)** - The traffic control and congestion control procedures for ATM. ATM layer traffic control refers to the set of actions taken by the network to avoid congestion conditions. ATM layer congestion control refers to the set of actions taken by the network to minimize the intensity, spread and duration of congestion. The following functions form a framework for managing and controlling traffic and congestion in ATM networks and may be used in appropriate combinations:

Connection Admission Control Feedback Control Usage Parameter Control Priority Control Traffic Shaping Network Resource Management Frame Discard ABR Flow Control

**Traffic Parameter** - A parameter for specifying a particular traffic aspect of a connection.

**Trailer -** the protocol control information located at the end of a PDU.

**Transit Delay -** the time difference between the instant at which the first bit of a PDU crosses one designated boundary, and the instant at which the last bit of the same PDU crosses a second designated boundary.

**Transmission Control Protocol (TCP)** - a specification for software that bundles and unbundles sent and received data into packets, manages the transmission of packets on a network, and checks for errors.

Transmission Control Protocol/Internet Protocol (TCP/IP) - a set of communications protocols that has evolved since the late 1970s, when it was first developed by the Department of Defense. Because programs supporting these protocols are available on so many different computer systems, they have become an excellent way to connect different types of computers over networks.

**Transmission Convergence (TC) -** generates and receives transmission frames and is responsible for all overhead associated with the transmission frame. The TC sublayer packages cells into the transmission frame.

**Transmission Convergence Sublayer (TCS) -** This is part of the ATM physical layer that defines how cells will be transmitted by the actual physical layer.

**Transparent Asynchronous Transmitter/Receiver Interface (TAXI) -** Encoding scheme used for FDDI LANs as well as for ATM; supports speed typical of 100 Mbps over multimode fiber.

**Transport Layer** - Layer Four of the OSI reference model that is responsible for maintaining reliable end-to-end communications across the network.

**trap -** a program interrupt mechanism that automatically updates the state of the network to remote network management hosts. The SNMP agent on the switch supports these SNMP traps.

**Trivial File Transfer Protocol (TFTP) -** Part of IP, a simplified version of FTP that allows files to be transferred from one computer to another over a network.

**Twisted Pair -** Insulated wire in which pairs are twisted together. Commonly used for telephone connections, and LANs because it is inexpensive.

**Unassigned Cells -** a generated cell identified by a standardized virtual path identifier (VPI) and virtual channel identifier (VCI) value, which does not carry information from an application using the ATM Layer service.

**Unavailable Seconds (UAS) -** a measurement of signal quality. Unavailable seconds start accruing when ten consecutive severely errored seconds occur.

**UNI 3.0/3.1 -** the User-to-Network Interface standard set forth by the ATM Forum that defines how private customer premise equipment interacts with private ATM switches.

**Unicasting -** The transmit operation of a single PDU by a source interface where the PDU reaches a single destination.

**Universal Test & Operations Interface for ATM (UTOPIA) -** Refers to an electrical interface between the TC and PMD sublayers of the PHY layer.

**Unshielded Twisted Pair (UTP) -** a cable that consists of two or more insulated conductors in which each pair of conductors are twisted around each other. There is no external protection and noise resistance comes solely from the twists.

**Unspecified Bit Rate (UBR) -** a type of traffic that is not considered time-critical (e.g., ARP messages, pure data), allocated whatever bandwidth is available at any given time. UBR traffic is given a "best effort" priority in an ATM network with no guarantee of successful transmission.

**Uplink** - Represents the connectivity from a border node to an upnode.

**Usage Parameter Control (UPC) -** mechanism that ensures that traffic on a given connection does not exceed the contracted bandwidth of the connection, responsible for policing or enforcement. UPC is sometimes confused with congestion management (see *congestion management*).

**User Datagram Protocol (UDP) -** the TCP/IP transaction protocol used for applications such as remote network management and name-service access; this lets users assign a name, such as "RVAX\*2,S," to a physical or numbered address.

**User-to-Network Interface (UNI) -** the physical and electrical demarcation point between the user and the public network service provider.

**V.35 -** ITU-T standard describing a synchronous, physical layer protocol used for communications between a network access device and a packet network. V.35 is most commonly used in the United States and Europe, and is recommended for speeds up to 48 Kbps.

**Variable Bit Rate (VBR) -** a type of traffic that, when sent over a network, is tolerant of delays and changes in the amount of bandwidth it is allocated (e.g., data applications).

**Virtual Channel (or Circuit) (VC) -** a communications path between two nodes identified by label rather than fixed physical path.

**Virtual Channel Connection (VCC)** - a unidirectional concatenation of VCLs that extends between the points where the ATM service users access the ATM Layer. The points at which the ATM cell payload is passed to, or received from, the users of the ATM Layer (i.e., a higher layer or ATMM-entity) for processing signify the endpoints of a VCC.

**Virtual Channel Identifier (VCI)** - the address or label of a VC; a value stored in a field in the ATM cell header that identifies an individual virtual channel to which the cell belongs. VCI values may be different for each data link hop of an ATM virtual connection.

**Virtual Channel Link (VCL)** - a means of unidirectional transport of ATM cells between the point where a VCI value is assigned and the point where that value is translated or removed.

**Virtual Channel Switch -** a network element that connects VCLs. It terminates VPCs and translates VCI values. The Virtual Channel Switch is directed by Control Plane functions and relays the cells of a VC.

**Virtual Connection -** an endpoint-to-endpoint connection in an ATM network. A virtual connection can be either a virtual path or a virtual channel.

**Virtual Local Area Network (VLAN) -** Work stations connected to an intelligent device which provides the capabilities to define LAN membership.

**Virtual Network Software (VINES) -** Banyan's network operating system based on UNIX and its protocols.

Virtual Path (VP) - a unidirectional logical association or bundle of VCs.

**Virtual Path Connection (VPC) -** a concatenation of VPLs between virtual path terminators (VPTs). VPCs are unidirectional.

**Virtual Path Identifier (VPI)** - the address or label of a particular VP; a value stored in a field in the ATM cell header that identifies an individual virtual path to which the cell belongs. A virtual path may comprise multiple virtual channels.

**Virtual Path Link (VPL) -** a means of unidirectional transport of ATM cells between the point where a VPI value is assigned and the point where that value is translated or removed.

**Virtual Path Switch -** a network element that connects VPLs, it translates VPI (not VCI) values and is directed by Control Plane functions. The Virtual Path Switch relays the cells of a Virtual Path.

**Virtual Path Terminator (VPT) -** a system that unbundles the VCs of a VP for independent processing of each VC.

**Virtual Private Data Network (VPDN) -** a private data communications network built on public switching and transport facilities rather than dedicated leased facilities such as T1s.

**Virtual Private Network (VPN) -** a private voice communications network built on public switching and transport facilities rather than dedicated leased facilities such as T1s.

Virtual Source/Virtual Destination (VS/VD) - An ABR connection may be divided into two or more separately controlled ABR segments. Each ABR control segment, except the first, is sourced by a virtual source. A virtual source implements the behavior of an ABR source endpoint. Backwards RM-cells received by a virtual source are removed from the connection. Each ABR control segment, except the last, is terminated by a virtual destination. A virtual destination assumes the behavior of an ABR destination endpoint. Forward RM-cells received by a virtual destination are turned around and not forwarded to the next segment of the connection.

**Virtual Tributary (VT)** - a structure used to carry payloads such as DS1s that run at significantly lower rates than STS-1s.

Warm Start Trap - an SNMP trap which indicates that SNMP alarm messages or agents have been enabled.

Wide-Area Network (WAN) - a network that covers a large geographic area.

**Wideband Channel -** Communications channel with more capacity (19.2K bps) than the standard capacity of a voice grade line.

**X.21 -** ITU-T standard for serial communications over synchronous digital lines. The X.21 protocol is used primarily in Europe and Japan.

**X.25** - a well-established data switching and transport method that relies on a significant amount of processing to ensure reliable transport over metallic media.

**Yellow Alarm -** An alarm signal sent back toward the source of a failed signal due to the presence of an AIS (may be used by APS equipment to initiate switching).

**Zero Byte Time Slot Interchange (ZBTSI) -** A technique used with the T carrier extended superframe format (ESF) in which an area in the ESF frame carries information about the location of all-zero bytes (eight consecutive "0"s) within the data stream.

**Zero Code Suppression -** The insertion of a "1" bit to prevent the transmission of eight or more consecutive "0" bits. Used primarily with T1 and related digital telephone company facilities, which require a minimum "1's density" in order to keep the individual subchannels of a multiplexed, high speed facility active.

**Zero-Bit Insertion -** A technique used to achieve transparency in bit-oriented protocols. A zero is inserted into sequences of one bits that cause false flag direction.

Glossary

## Index

A	monitoring B - 1
AMI 5 - 8	SNMP configurationB - 1
ATM Management Interface (AMI) 5 - 8	topology
ATM switch element managers 5 - 8	synchronization 4 - 19
С	topology data tablesB - 5
commands	fv_address_1
summary 4 - 28	fv_container_1
·	fv_endpoint_1 B - 11
D	fv_interface_1 B - 10
database	fv_link_1 B - 11
fvovmon	fv_node_1
fvtrapd	fv_sig_entity_1 B - 12
fvtrimdb	supporting tables B - 13
how information is stored 4 - 11	discovery interval 4 - 5
maintaining the database A - 7 mapping data tables B - 23 fv_mapping_1 B - 23	F ForeView architecture
supporting tables	dependencies on ForeThought         software versions       1 - 4         getting started       3 - 1         GUI       1 - 2         integrated processes       3 - 1         integration with HP OpenView       1 - 3
fv_node_s_1       B - 18         fv_spans_link_s_1       B - 21         fv_uni_link_s_1       B - 20         supporting tables       B - 21	
non-FORE devices	on-line manual
schema overview	Fore View 4.3 tools with Fore View 5.0 $2-7$ Fore View 4.3, removing $2-8$ Fore View components, overview $3-2$ Fore View licensing $2-6$
mapping B - 1	Fore View, run commands       3 - 3         fvaddlink       4 - 13

## Index

fvdbtrim 4 - 10, A - 7	copy and paste 5 - 13
fvdisco	creating location symbols 5 - 11
fvmon	fvaddlink 4 - 14
fvovmon	fvtopomap4 - 16, 4 - 17
fvseed	interswitch connections5 - 4, 5 - 10
fvtopomap	launching switch managers 5 - 8
fvtrapd	managing objects 5 - 14
н	managing symbols and maps $\ldots5$ - $11$
hardware requirements 2 - 3	non-FORE devices 4 - 13
	physical vs. logical topology 5 - 2
I	root level map 5 - 3
importing host name file 2 - 14	status colors 5 - 10
L	switch connections submap 5 - 5
licensing agreements	switch fabrics submap 5 - 6
М	NIAMs, defined 3 - 2
monitoring	0
fvmon4 - 6, 4 - 10	obtaining software from TACtics online . 2 - 5
link status 4 - 8	obtaining the software
node status	OpenView integration
nodes and links 4 - 6	fvovmon 4 - 16
polling SNMP agents 4 - 6	fvtopomap 4 - 16
status monitoring NIAM (fymon) . 4 - 6	fvtrapd 4 - 16
<u> </u>	P
N	processes
network discovery	fvaddlink 4 - 13
device information 4 - 3	fvdbtrim 4 - 10
discovery NIAM (fvdisco) 4 - 2 fvdisco 4 - 5	fvdisco 4 - 5
fvseed	fvmon 4 - 10
ILMI SNMP proxy (ISP) MIB 4 - 2	fvovmon 4 - 11, 4 - 16, 4 - 21
LAN access devices	fvseed 4 - 4
seeding the discovery 4 - 4	fvtopomap4 - 16, 4 - 17
Network Information Access	fvtrapd4 - 11, 4 - 16
Modules (NIAMs), defined 3 - 2	fvtrimdb
network mapping	
ATM networks submap 5 - 4	
bidirectional connections 5 - 7	

R	checking error logs
run commands 3 - 3	checking fvtopomap
commands for Windows NT	on UNIX
services 3 - 5	on Windows NTA - 3
fvsrvc -start 3 - 5	checking processes
fvsrvc -status 3 - 5	controlling NIAM startup
fvsrvc -stop 3 - 5	at reboot
using the services control panel 3-	deleting <i>ForeView</i> map objects A - 5
5	examining the ovw database $\dots A$ - $4$
UNIX commands for daemons 3 - 4	maintaining the database $\dots A$ - 7
fvstart 3 - 4	modifying launch environments A - 8
fvstatus 3 - 4	W
fvstop 3 - 4	web link for technical manuals 2 - 6
S	Windows NT services
SNMP configuration 4 - 27	vindovis ivi scrvices
software installation	
from a CD-ROM 2 - 12	
from a TACtics online file 2 - 10	
importing host name file 2 - 14	
obtaining software from	
TACtics online 2 - 5	
obtaining the software 2 - 5	
on UNIX platform 2 - 12	
on Windows NT platform 2 - 13	
system requirements 2 - 3	
software requirements 2 - 4	
system requirements	
т	
technical manuals web link 2 - 6	
tenacity	
trap handling	
FORE ATM-specific 4 - 16	
fvtrapd	
supported traps 4 - 22	
trap destination table 4 - 22	
troubleshooting	

Index